(2)

US ARMY
LABORATORY COMMAND

CONTRACT REPORT NUMBER 7-92
PREPARED FOR THE

HUMAN ENGINEERING LABORATORY

BY

CHARLES RIVER ANALYTICS, INC.
55 Wheeler Street
Cambridge, MA 02138

DESIGN FOR TACTICAL SITUATION AWARENESS DISPLAY

June 15, 1992

Contract No. DAA15-90-C-0019

US ARMY HUMAN ENGINEERING LABORATORY
HEL
ABERDEEN PROVING GROUND, MARYLAND

Approved for public release;
distribution unlimited.

ABERDEEN PROVING GROUND, MARYLAND 21005-5001

Charles River Analytics Inc.

Final Report No. R89351
Contract No. DAA15-90-C-0019

Design for Tactical Situation Awareness Display

Greg L. Zacharias and Paul G. Gonsalves
Charles River Analytics Inc.
55 Wheeler Street
Cambridge, MA 02138

June 15, 1992

Prepared for:

Dr. Richard A. Monty
Aviation & Air Defense Division
U.S. Army Human Engineering Laboratory
Aberdeen, MD

ACKNOWLEDGMENT

This work was performed under U.S. Army Contract DAA15-90-C-0019 with the Aviation and Air Defense Division of the U.S. Army Human Engineering Laboratory, Aberdeen, MD. The authors thank the Contracting Officer's Representative (COR) Dr. Richard A. Monty for his support and technical input through the course of this project. We also thank Ms. Virginia E. Mahoney for report preparation and revision.

TABLE OF CONTENTS

1. INTRODUCTION

Because of continuing advances in on-board sensors, computers, and displays, we can expect to see significant growth in the amount of information presented to rotorcraft crews by next-generation tactical situation displays. If unchecked, this growth will easily exceed the crew's capability for dealing with the available on-board information base in a timely and effective fashion. A range of technologies are now being proposed to provide appropriate levels of information management and situation awareness, ranging from hardware upgrades like touch screen displays, to advanced computing horsepower to support multi-sensor data fusion, to new AI technologies including expert systems like Pilot's Associate. The problem, however, is not that there is a dearth of candidate technologies; rather it is that we have no reliable means for evaluating them in terms of the fundamental situation awareness they afford the crew. What is called for is an objective metric for evaluating the awareness level provided a crew by a new cockpit technology, and a rational means for using this metric in a structured methodology involving: 1) identification of mission-specific tactical informational requirements; 2) generation of candidate display formats and symbologies; and 3) metric-based evaluation of afforded situation awareness.

There are several requirements which must be met if we are to develop a successful metric. First, the metric must account for the pilot's fundamental capabilities and limitations in processing and acting upon information, starting with his performance in sensory/perceptual processing and data fusion, proceeding to his strategies for situation assessment and decision-making, and following through to his execution of a range of procedural activities. Second, the metric must be applicable to a broad variety of display technologies, and should support evaluations of widely differing display concepts and formats; there should also be provision for growth and modification of the metric, as cockpit configurations evolve. Third, the metric should account for other system-related design factors which impinge upon mission performance, such as rotorcraft maneuverability, weapons lethality, etc., and there should exist a means of incorporating these factors in a design evaluation in a relatively straight-forward fashion. Finally, the rationale behind the metric should be clear to the user, the display designer, and the method of application should be relatively free of subjective design decisions, to ensure user acceptance and consistent design evaluation results.

These requirements eliminate from consideration a number of potential approaches to developing the required metric. The need to consider the human's full range of information-processing activities, from sensory processing to procedure execution, argues against a conventional human factors assessment, in which one considers only the basic display attributes, such as legibility and *busyness*, and ignores the more critical issues regarding information

transfer and level of abstraction. The requirement for applicability to a range of display technologies rules out the use of a design handbook approach, since development of such technologies almost always outpaces handbook-type guideline development. The need to account for non-cockpit systems-related design factors suggests that a pure information-coding analysis of the display is likely to miss critical couplings between the external environment, the vehicle and its subsystems, and the tactical situation assessment needs of the crew. These couplings, will, in the ultimate, determine the effectiveness of any proposed tactical situation display.

We believe that these limitations can be circumvented by the use of an awareness metric founded on an integrated model of the overall crew/vehicle system. Such a model can provide the critical information-based linkage between the external environment, the vehicle, the tactical situation display, and the crew. It can serve as a framework for integrating the pilot's perceptual data base with his procedural knowledge, to provide insight as to how situations are assessed by the pilot and re-assessed in light of new information. Finally, a model-based approach provides us with the means of generating an explicit representation of the pilot's internal assessment of the situation. Direct comparison with the actual situation lets us define a metric of situational disparity, and its inverse, situation awareness. This explicit model-based definition not only provides a clear path to empirical measurement and validation for the developer, but it also supports direct display evaluation for the user.

1.1 Technical Objectives

The primary objective of the effort is to evaluate the feasibility of developing mission-specific situation awareness (SA) display requirements, using a model-based approach for objective evaluation of candidate designs. The approach integrates a pilot/vehicle model of the overall system, a metric of situation awareness, and a demonstration of display design specification and evaluation. Basic questions addressed during the effort are:

- Can we integrate a functional model of the situation assessment process with an appropriate metric, to provide a measure of the situation awareness afforded by a given display or display format? What is the potential for the range of display options and decision aids that might be evaluated, and what is the scope of the information base we might expect to evaluate via this metric?
- Can we identify enabling technologies needed for the development of this model-based approach, and propose appropriate means for representing the crew's fundamental functions of information processing (IP), situation assessment (SA), and decision-making (DM)? What is an appropriate metric of SA, and can it be used to support objective evaluations of SA display design?

- Can we demonstrate the use of this metric in a design example involving a candidate situation awareness display? How can we formalize the process of: analyzing mission-specific informational requirements, generating candidate display formats and symbologies, and evaluating them via the proposed metric? How does a metric-based ranking of candidate display aids compare with a subjective assessment of display effectiveness?

- What are the limitations of the proposed model-based approach, and how can they be eliminated or reduced? What are recommended development paths to improve the design approach, validate its predictive utility, and encourage its use in the display design community?

In our initial effort, we have focused on the development, implementation and a proof-of-concept demonstration of a model-based method for specifying situation awareness display requirements. For a follow-on effort, we plan for the development, demonstration, and evaluation of a prototype tactical display design and evaluation tool.

1.2 Technical Approach

Our general technical approach to evaluating feasibility of the SA display analysis design method centers on: 1) specification of an integrated functional model of the crew/system that represents the system relations, the pilot activities, and provides "hooks" to the crew's internal states, assessed situations, and decisions; 2) assessment of enabling technologies for crew/system model development, including modern estimation (ME) techniques, artificial neural networks (ANNs), and expert systems (ES) technology; 3) implementation of a limited scope version of the model for a display design evaluation and assessment of afforded SA; 4) conduct of a proof-of-concept demonstration to evaluate SA as a function of display format and aids; and 5) identification of design tool objectives and development paths for full-scope implementation and validation.

We specify an integrated functional crew/system model (CSIM) that allows us to combine and integrate system-related and human-related components that drive overall performance and crew SA during an engagement. This model provides an architecture for integrating the crewmembers' basic functions of: 1) sensory/perceptual processing of the display interface cues; 2) information processing (IP) of continuous vehicle states and discrete cueing aids; 3) situation assessment (SA) for driving engagement-relevant decisions; 4) decision-making (DM) for selecting among alternative actions; and 5) procedure execution for effecting guidance and missile firing commands.

We describe a narrow range of crew procedures that might be evaluated within this framework, and identify candidate model-based SA metrics for display evaluation studies.

In the specification of the CSIM model, we evaluate enabling technologies for full-scale development. For the sensory processing, information fusion, continuous control, and discrete detection components of the model, we propose the use of modern estimation (ME) techniques. For the situation assessment component of the model, we propose the use of artificial neural networks (ANNs). For the decision-making, procedure selection, and procedure-effecting components of the model, we propose the use of expert systems (ES) technology. We evaluated the applicability of each enabling technology to CSIM component implementation and design tool development.

We implement a limited-scope version of the crew/system model to support a demonstration of its use in evaluating a display proposed for tactical SA. The task chosen is a rotorcraft tactical engagement in a limited air-to-air engagement. Major tasks facing the crew are: identification friend or foe (IFF), prioritization of threat targets, and fire point selection (FPS). The analysis effort begins with the development of both ownship and threat/friendly dynamic models, the development of a missile model, and the development of simplified radar models to provide relative range and velocity measurements to objects in the tactical area. The radar display is modeled on a Tactical Situation Display (TSD) for Rotorcraft Counter-Air Engagements (RCE) developed by HEL. The representation effort focuses on modeling the information content of the display, the generation of derived states to feed a model of the crew's situation assessment function, the specification of a rule-base for decision-making, and the formalization of a procedure effector for guidance, control, fire-point selection, and other procedures. A nominal analysis is conducted with a baseline TSD display, and variants on the basic format are evaluated via model-based analysis. In addition, a number of decision aids are evaluated regarding their effectiveness in providing the crew with improved situation awareness (SA). In particular, four decision aids are evaluated: one to aid in the IFF function, one to aid in target prioritization, one for fire-point selection, and the last for TSD attention focusing. Of primary interest here is the degree of SA provided to the crew by changes in display format and/or changes in decision aid implementation.

1.3 Summary of Results

The primary result of this study is a successful proof-of-concept demonstration of the model-based approach to evaluating situation awareness display requirements. The major study findings supporting this demonstration effort can be summarized as follows.

. The specification of an overall crew/system model provides us with a development framework for evaluating the impact of display characteristics on crew awareness, and conversely, the impact of crew activity on system performance. It is an interactive framework that represents both the display characteristics and the crew procedures for dealing with the information in that display. It provides for an explicit representation of the crew's fundamental functions of information-processing (IP), situation assessment (SA), and decision-making (DM). It supports the development of objective metrics of performance and SA, and the subsequent evaluation of competing display formats and decision aids.

The evaluation of enabling technologies focuses on three areas: modern estimation (ME) models of the crew's IP function; artificial neural network (ANN) representations of the crew's SA functions; and expert system (ES) implementations of the crew's DM activities.

The modern estimation (ME) modeling work centers on the optimal control model (OCM) of the pilot/vehicle system. It is a standard tool for closed-loop evaluation of pilot performance under different monitoring assumptions and control options, and has been used extensively in pilot/vehicle analysis. For this effort it is used to model the *front-end* processing functions of the pilot, specifically: attention allocation amongst the available displays, state estimation for the generation of continuous system states, and discrete event detection. In the modeling effort, it is assumed that the pilot has a close matched *internal model* of the external environment. However, in an extended configuration of the OCM, it is possible to evaluate task performance where there are distinct mismatches between the pilot's *internal model* of the external environment and the environment itself. This provides the potential for evaluation of display configurations and aids in cases where significant mismatches can be expected due to prior misconceptions, on the part of the pilot regarding the system dynamics.

The evaluation of artificial neural networks (ANNs) for representing the situation assessment behavior of the crew focuses on the pattern recognition capabilities of ANNs. ANNs can provide general pattern recognition capabilities in the general state/event space generated by the CSIM information processor (IP), thus supporting the recognition of predefined *situational patterns* occurring over both space and time. Modeling of the pilot's SA function in this manner calls on a number of strengths afforded by ANNs. First, ANNs are particularly effective in implementing non-verbal pattern recognition functions which are algorithmically ill-defined and

highly perceptual in nature. Second, ANNs can provide for considerable data compression with a many-on-few topological mapping from several state/event inputs to a few situational outputs. Third, the node normalization of ANN activation functions can be used to provide classification probabilities in a manner analogous to Bayesian discriminant analysis. Finally, an ANN model has the potential for modeling on-line unsupervised learning of tactically relevant patterns, and the sharpening of tactical skills that comes with repeated successful engagements. In this effort we focus on modeling the crew's recognition of threatening patterns, to support the IFF function.

The evaluation of expert system (ES) representations of the crew's decision-making behavior centers on data-driven procedural activities. We believe that the potential utility of an ES representation rests on four major factors. First, there is a natural mapping of the ES architecture to the crew DM functions; the ES data base represents the assessed situation, the ES rule base reflects the crew's procedure set, and the ES inference engine models the crew's DM algorithm(s). Second, an ES implementation can readily incorporate heuristics employed by the crew to *short-cut* extensive decision/procedure sequences, in accord with the stimulus-driven activity path postulated by several researchers. Third, ESs can implement dynamic updating of their knowledge base, to account for short-term trends peculiar to the mission phase or task. Finally, ESs can serve as a knowledge repository for implementation of the *upstream* crew functions of information processing (IP) and situation assessment (SA). In particular, ESs can maintain a knowledge base of IP techniques and modes incorporated in the modern estimation (ME) formulation used to model the crew; likewise, ESs can characterize SA strategies and procedures represented in an ANN formulation of the SA function, via a specification of network topologies and weighting schemes used for different SA objectives during a mission. In this effort we focus on modeling the crewmember's decision functions for threat/friendly discrimination, target prioritization, and fire-point selection.

The proof-of-concept demonstration focuses on a rotary-wing air-to-air engagement and provides: a basis for evaluating the requirements for problem setup; supports an objective evaluation of display formats and decision aids; and provides the foundations for a detailed evaluation of situational awareness during the engagement. The major findings of our demonstration effort can be summarized as follows:

- Display analysis and problem set-up is straightforward, and does not require an extensive procedural data base. Analysis of information content and format follows standard model-based approaches. Decision aid enhancements can be incorporated in a straightforward fashion through event generation and processing.

- Monitoring and control strategies on the part of the pilot can be explicitly defined. Attention allocation strategies used for processing of display information can be specified, as well as control strategies for guidance and fire-point selection.
- A variety of metrics can be evaluated with this approach. An overall metric representing the crew's SA can be computed both as a function of time and over the course of the engagement. In addition, metrics regarding state estimation accuracy can be computed, as well as engagement performance. Tracking of engagement progress can be effected via internal metrics. Enhancement-induced differences are reflected across metrics as well.
- The approach also provides a means of evaluating the specifics of the scenario (i.e., initial engagement geometry) on effectiveness of enhancements. In particular, enhancement effectiveness can be evaluated in low versus high-tempo engagement scenarios.
- The effect of display format variations focuses on the impact of display resolution and map scale range. The analysis shows that SA is effectively on the knee of the curve, in that a lower resolution TSD results in significant decreases in SA, whereas a higher resolution TSD yields no effective improvement in SA.
- The effort demonstrates how use of an IFF display aid can result in a 50% increase in SA, as crewmember confidence on the display aid increases to 100%. A general linear progression in SA occurs with increasing confidence and use of the aid.
- The study with the priority display aid shows there is little effect under the normal scenario; however, when the number of threat/friendlies is doubled in a higher tempo scenario, a 15% improvement in SA is seen.
- The study of the FPS display aid also displays little effect in the normal low-tempo scenario; however, a 40% improvement is seen in SA for the higher tempo scenario, with an approximately linear improvement in SA as confidence and use of the aid is increased.
- The study of the attention-focusing aid shows little effect when focusing is used to enhance a display. In fact, a slight negative effect is found due to the lack of information provided for low priority targets.

These results and others generated under this effort demonstrate that reasonably realistic activity traces can be generated with a fairly limited procedure rule base, and that the dynamic

evolution of the pilot's situation assessment, decision-making, and task performance can be followed throughout an engagement. The results also show how simple metrics can be generated for internal model states related to SA and to external world states, to support evaluations of situation awareness and the effectiveness of display formats and decision aids.

Finally, the evaluation effort demonstrates how a number of engagement and crew variables affect engagement outcome. We can directly study the effects of: 1) initial engagement geometry; 2) sensor inaccuracies; 3) pilot-related limitations in sensory/perceptual processing; 4) internal state estimation strategy; 5) pattern recognition capabilities in situation assessment; and 6) decision-making procedure definitions. A wide variety of factors can be evaluated over a range of tactically relevant values.

1.4 Report Outline

This report summarizes and documents our proof-of-concept demonstration of a model-based method for specifying situation awareness display requirements.

Chapter 2 describes the architecture for a crew/systems integration model (CSIM) to be used as the basis for optimizing task allocation across crewmembers. Section 2.1 presents the general model requirements, while section 2.2 describes the CSIM model designed to meet these requirements.

Chapter 3 describes three general enabling technologies for development of the crew/system integration model (CSIM). Section 3.1 describes our approach to modeling the crew's information-processing (IP) functions in a modern estimation (ME) framework, using the optimal control model (OCM) of the human pilot. Section 3.2 then presents an overview of artificial neural networks (ANNs), as background for the development of the situation assessment (SA) submodel of CSIM. Section 3.3 reviews Expert Systems (ES) technology, to support the development of a model of the crew's decision-making (DM) functions. Finally, in section 3.4, we discuss our commercially-available hybrid neural network expert system tool *NueX* used for neural network situation assessment model development and knowledge-based decision-making.

Chapter 4 describes an implementation of the crew/system integration model (CSIM) for use in a model-based analysis of a specific situation display. Section 4.1 describes the overall system implementation. Section 4.2 details the external and vehicle modules including threat and ownship and subsystem models. Section 4.3 completes the chapter with a description of the crew-centered portion of the modeling effort.

Chapter 5 presents the results of our proof-of-concept demonstration of the model-based method. It focuses on the analysis of a tactical situation display. Section 5.1 presents results for a baseline scenario and a nominal baseline display. Section 5.2 presents results for variations in the baseline format and for several different display enhancement options.

Chapter 6 concludes the report with a summary, conclusions, and recommendations for future work.

2. CREW/SYSTEM INTEGRATION MODEL

In this chapter we describe the architecture for a crew/systems integration model (CSIM) to be used as the basis for specifying situational awareness display requirements. Section 2.1 presents the general model requirements, while section 2.2 describes the CSIM model designed to meet these requirements. Section 2.3 concludes with a discussion model-based metrics for situation awareness.

2.1 Requirements for Integrated Crew/System Model

An integrated crew/system model must account for the crew, the vehicle, and the mission environment. If it is to be effective, it should support the systematic exploration of issues revolving around cockpit interface design, procedure specification, and crew workload. Several general implications for modeling can be identified:

- A system model is needed; one that accounts for the interactions of crew, procedures, vehicle, mission plan, and threat environment.
- A means must be provided for modeling the essential function of data fusion, and the allocation of attention among a disparate number of information sources providing partial, time-varying, and sometimes erroneous information to the crew.
- The issues of interest revolve around information transfer within and outside the cockpit, and include situation assessment, decision-making, and procedure execution; any model must therefore make explicit representation of these functions.
- There exists a requirement for providing flexibility in model representation to be able to deal with sometimes ill-defined functions, such as situation-assessment and decision-making under uncertainty, as well as the highly structured standard *by the book* procedures.
- Communication among crewmembers, and between the crew and outside command structures, must be accounted for at least with respect to the basic information transfer provided, and the task workload imposed on the crew.
- Finally, the model should provide a quantitative basis for evaluating situational awareness to support rationally-based evaluations of selected display format and decision aids.

These modeling requirements have led to a succession of model development efforts and model conceptualizations. The modeling approach has its roots in the development that began

with the Optimal Control Model (OCM), described by Kleinman and Baron (1971). This is an information-processing model of the operator of a dynamic system, grounded in modern control and estimation theory, which accounts for closed-loop man/machine performance across a range of primarily continuous control tasks (e.g., flight-path control). We will describe this model in detail in the next chapter. Building on this model, the dynamic decision-making model (DDM) of Pattipati, et al. (1979), Pattipati, et al. (1982), and Pattipati, et al. (1983) was developed to account for the operator's discrete decision-making performance in a generic multi-task supervisory control environment.

The first attempt to integrate continuous and discrete information processing with the structured procedural activities of the flight crew was with PROCRU, a model developed to evaluate commercial approach procedures during landing (Baron, et al. (1980), Milgram, et al. (1984), Visser (1988)). A number of approach simulations were conducted under different procedural assumptions, to support evaluations of crew information-processing requirements, performance, and workload. An upgraded version was also developed to model the anti-aircraft (AAA) crew, and to evaluate performance against targets using a range of defensive countermeasures (Zacharias, et al. (1982)). Here, the task of situation assessment was first made explicit, and linked to procedurally-driven rules of engagement. Finally, we note that an enhanced model, CSIM, has been proposed for analysis of the fighter/attack mission (Zacharias and Baron (1982)) and has been used recently in an air superiority modeling effort (Zacharias (1989c)). This model provides a greater structural formalization of many of the functions included in previous versions; it also makes explicit some of the fundamental operational modeling requirements needed for realistic crew/systems analysis.

2.2 Crew/System Integration Model (CSIM)

Figure 2.1 provides an overview block diagram of the crew/system integration model (CSIM). The block diagram is broken into three major portions: one dealing with the external world, one with the aircraft itself, and one with the crewmember. Modeling of additional crewmembers is accomplished by essentially duplicating the lower (pilot) portion of the diagram for each additional crewmember, and modifying the list of procedures appropriately, to reflect the allocation of tasks among crewmembers. We now describe this single crewmember diagram in more detail.

The **external world models** block is meant to account for the important external drivers in, for example, an air-to-air engagement, which, in one way or another, serve to drive the rotorcraft and its crew during the course of the mission. On the basis of past mission analyses (e.g., Zacharias and Baron (1982)), we can identify four such drivers: 1) the targets themselves,

since they provide the basic tactical objective, and serve to drive the visual and/or sensor-aided target detection function; 2) any threat array defending the targets and the route to it, since it serves to determine mission route selection and also serves as a driver for visual and/or sensor-aided threat detection and countermeasure selection; 3) the weather since it serves to drive the vehicle dynamics (via winds and gust) and since it can limit visibility and terrain feature acquisition; and 4) the terrain, since it determines elevation of important terrain features, and since it serves to drive any visual and sensor-aided tasks.

Two primary communication paths, between the external world models and the two rotorcraft submodels (the vehicle dynamics and the set of subsystem models) are likely to serve as the primary pathway for non-VFR interactions between the rotorcraft and the scenario environment. We have indicated one other path, from the external world model block to the crewmember's extra-cockpit visual sensory channel, which supports direct visual observation of relevant external world factors (e.g., terrain height, target location, etc.).

The **rotorcraft** portion of the block diagram consists of three major components: the vehicle dynamics model, the set of (non-display) subsystems whose performance is particularly relevant to mission performance during the flight phases of interest, and the set of displays (also subsystems) which serve to present relevant vehicle and subsystem state information to the crewmember. As the diagram indicates, vehicle state can influence subsystem state, and vice versa. For example, target range (a function of vehicle and target position) can impact noise levels associated with radar sensor performance (subsystem operation); conversely, weapons release (subsystem state change) can directly affect the lift/drag coefficients of the vehicle, thus changing the vehicle dynamics. The diagram also indicates that all three rotorcraft blocks (dynamics, subsystems, and displays) can be driven by pilot-generated controls (e.g., via continuous stick commands into the dynamics, via operating mode discretes into the various subsystems, via display mode choices for programmable displays, etc.).

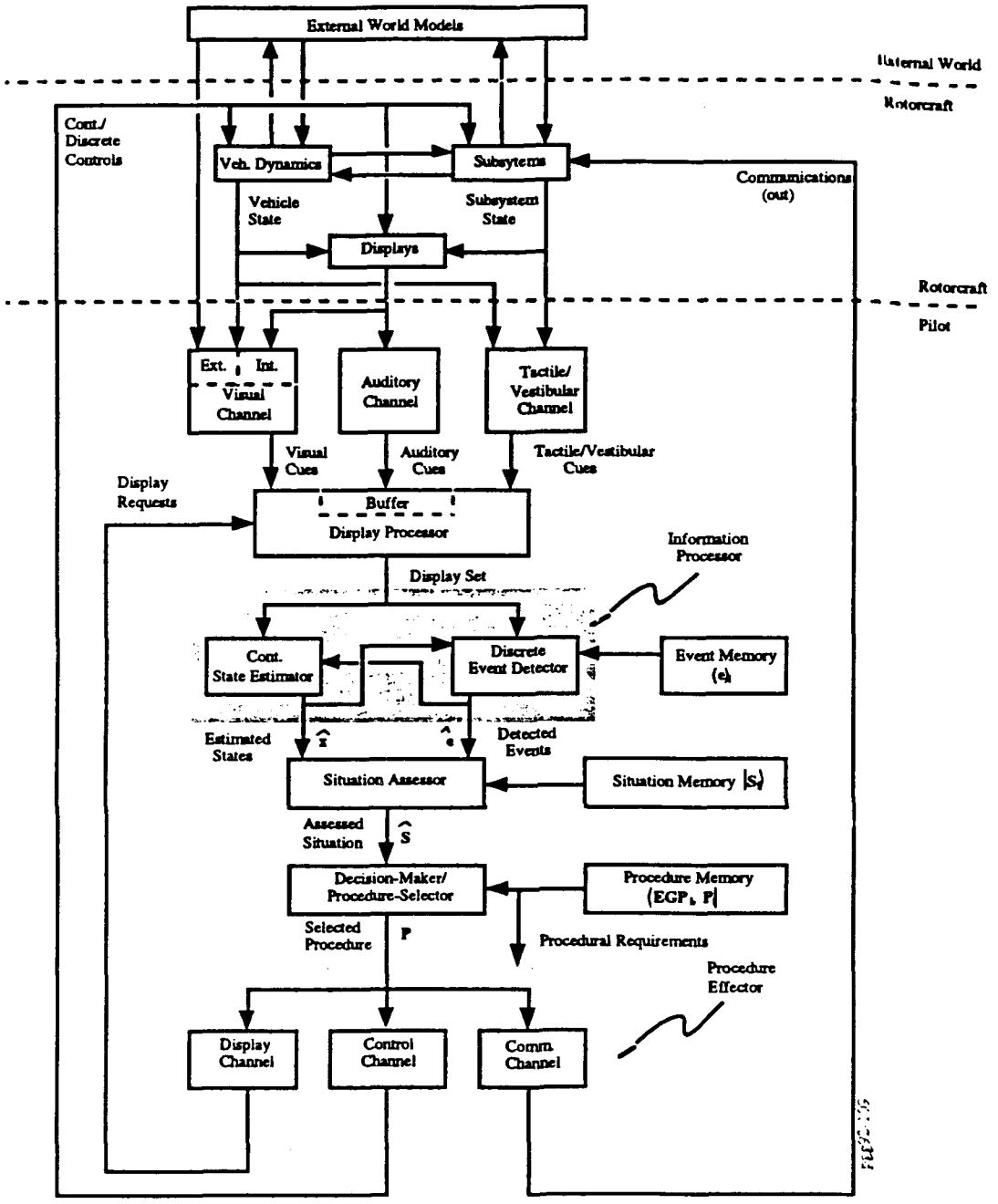


Figure 2.1: Crew/System Integration Model (CSIM)

All three major blocks of the rotorcraft portion serve to drive the crewmember's **sensory channels**, as shown in the overall block diagram. The vehicle state, in conjunction with the terrain features generated by the external world, serve to drive the crewmember's extra-cockpit visual channel, and provide him with terrain-relative flight path information. The rotorcraft displays, driven by both the dynamics and the subsystems (e.g., the altimeter driven by the vehicle state, or, say, a tactical situation display driven by the radar), serve to drive both visual

and auditory channels. We also have a *tactile/vestibular* channel, to account for non-visual and non-auditory cues picked up by the crewmember, such as rotational accelerations, blade slap, etc.

The **display processor** submodel accounts for the crewmember's sensory limitations as well as for monitoring decisions on how to allocate attention among competing sources of information. Visual sensory limitations are modeled in the same manner as in the OCM (Kleinman and Baron (1971)); Baron and Levison (1977)), except that the perceptual delay is neglected. An observation noise and a threshold are associated with each observed visual quantity. The thresholds are particularly important for external visual scene perception, for example, in limiting the quality of available vertical guidance information. Monitoring decisions must reflect the fact that the crewmember cannot process all sources of information simultaneously and must, therefore, decide which display to *attend to*. In the case of visual information, there is a fundamental choice as to where to fixate, on the external world or within the cockpit, and, if within the cockpit, on which portion of which in-cockpit display; with the current trend towards nested menu-driven displays, the issue of attention allocation becomes particularly important. We also assume that auditory information (after short-term buffering) similarly *competes* with visual information for pilot attention.

The **information processor** submodel consists of two submodels, a **continuous state estimator** and a **discrete event detector**. The estimator can be identical to that used in the OCM, a time-varying Kalman filter designed to generate optimal estimates of the current vehicle/system state; additional details are given in the following chapter. The internal model for the estimator changes with changes in vehicle dynamics due to changes in flight conditions, configuration changes (e.g., stores release), or changes in gust characteristics.

The outputs of the **estimator** are the estimate of the vehicle/system state, $\hat{\mathbf{x}}$ and the covariance of the estimation error, Σ. Such states would include the normal complement of vehicle linear and angular velocities, position, and attitude, as well as significant subsystem states, and states of any targets or threats that might influence situation assessment and procedure/task execution. With the assumption that the probability distribution for $\mathbf{x}$ is normal, the estimator produces status information, $\hat{\mathbf{x}}$, needed for vehicle control, as well as subjective probability estimates that can be used for event detection (ED) and situation assessment (SA). The error covariance Σ is also a measure of the crewmember's uncertainty in the estimate $\hat{\mathbf{x}}$, and can serve to influence monitoring decisions.

The **event detector** (ED) generates occurrence probabilities of mission relevant events, as perceived by the crewmember on the basis of his dynamically-changing information base. The event may be a failure (that did or did not result in an alarm), a request for action (say from

another crewmember), a mission-related milestone (e.g., crossing the FEBA), or some annunciated condition (e.g., radar lock-on). The inputs to the event detector are visual and auditory discretes picked up by the display processor, and state estimator outputs. The state information is used to detect state-related events, such as coming within range of a threat envelope. Note also the reliance on a memory list of possible events.

Sophisticated models exist for certain types of state-based failure detection, and these could readily be incorporated in the discrete event detector submodel block. For example, one could build upon the model proposed by Gai and Curry (1976), in which a generalized likelihood ratio (GLR) test is used to process optimal estimator residuals to determine the occurrence of a process failure event. Or, one could follow up on the modified GLR approach by Wewerinke (1981), or the discriminant analysis approach to discrete event detection proposed by Greenstein (1979). It suffices to note here that a range of options exist for modeling event detection performance, and for generating the desired event occurrence probabilities.

The **situation assessor** (SA) block takes in the estimated states $\hat{\mathbf{x}}$ and the detected events $\hat{\mathbf{e}}$, and generates an assessed situation state $\hat{\mathbf{S}}$, which is a multi-dimensional vector defining the occurrence probabilities of the possible situations facing the crew. For model tractability, we assume a fixed and pre-defined set of candidate situations, determined solely by their task relevance. That is, a situation defines an aggregated set of states, events, and possibly other situations which call for a given course of action, or procedure execution. In simple terms, a situation is a predicate that must be satisfied to activate a procedure in a production rule system. Naturally, both the situation and production rule activities can become quite complex, depending on the task being modeled.

Previous modeling efforts have used a number of approaches to modeling the situation assessment function. For simple flight crew procedures, situation assessment has been triggered directly on event detection, so that a detected event becomes an assessed situation (Baron, et al. (1980), Milgram, et al. (1984)). More complex situation assessment tasks have been modeled using Bayesian estimation techniques, where situational probabilities are computed on the basis of estimated states, detected events, and their corresponding statistics (Zacharias, et al. (1982)). Recent approaches have begun to recognize that accurate situation assessment relies heavily on recognizing critical *patterns* in a task-relevant state space. The potential for modeling this type of pattern recognition function via an Artificial Neural Network (ANN) structure appears particularly attractive. In an ANN context, the estimated states and detected events serve to drive the input nodes of the network, and generate output node activity patterns which define the assessed situation. Network topology and reinforcement paradigms can be selectable based on

mission phase or task, to support a wide range of SA functions. We discuss these options further in the next chapter.

The **decision maker** and **procedure selector** block takes in the assessed situation state $\hat{\mathbf{S}}$, and generates a selected task or procedure **P**, defined in the procedure memory shown. The definition of these procedures is an essential step in any task modeling effort, and it is important to note that the term procedure can apply to tasks in general; a procedure in these terms can have considerably more cognitive content than might normally be considered.

We assume that the crewmember knows what is to be done and, essentially, how to accomplish the objective. However, he must decide what procedure to do next. This is a decision among alternatives and the procedure selected is assumed to be the one with the highest expected gain for execution at that time. The Expected Gain of a Procedure, (EGP), is a function that is selected to reflect the urgency or *priority* of that procedure as well as its *value*, in terms of its contribution to overall task completion.

For procedures that are triggered by the crewmember's situation assessment related to system state, the EGP functions can be appropriate subjective probabilities. Procedures that are triggered by detected events, such as in-cockpit alarms, etc., can be characterized by EGP's that are explicit functions of time. For either type of function, the EGP will increase, following the perception of the triggering event or situation, until the procedure is performed, or until a time such that the procedure is assumed to be *missed* or no longer appropriate for execution. The EGP functions are chosen so that it is possible to establish a default procedure for each crewmember.

The decision maker/procedure selector block is particularly amenable to modeling via an expert systems (ES) approach. In an ES context, the data base is comprised of the assessed situation, the rule base is comprised of the in-memory procedure rules, and the inference engine is the decision-making algorithm. Thus, we can make full use of well-known and effective ES inferencing strategies, as discussed in the next chapter.

The selection and execution of a procedure will result in an action or a sequence of actions. Three types of actions are considered: **control actions**, **display requests**, and **communications**. The control actions include continuous manual flight control inputs to the aircraft and discrete control settings. Display requests result from procedural requirements for specific information and, therefore, raise the attention allocated to the particular information source. Communications are verbal requests or responses as demanded by a procedure, and are modeled directly as the transfer of either state, command, or event information.

3. ENABLING TECHNOLOGIES FOR CSIM DEVELOPMENT

We now describe three general enabling technologies for development of the crew/system integration model (CSIM) we have just described. Section 3.1 describes our approach to modeling the crew's information-processing (IP) functions in a modern estimation (ME) framework, using the Optimal Control Model (OCM) of the human pilot. Section 3.2 then presents an overview of artificial neural networks (ANNs), as background for the development of the situation assessment (SA) submodel of CSIM. Section 3.3 reviews Expert Systems (ES) technology, to support the development of a model of the crew's decision-making (DM) functions. Finally, in section 3.4, we discuss our commercially-available hybrid neural network expert system tool *NueX* used for neural network situation assessment model development and knowledge-based decision-making.

3.1 Optimal Control Model (OCM) for Information Processing

Our approach to modeling the crew's information-processing (IP) functions centers on the Optimal Control Model (OCM) of the human pilot, a model which has been developed within the systems framework of modern estimation (ME) and control theory (Kleinman, et al. (1971)). The basic assumption underlying the model is that the well-trained, well-motivated human operator behaves optimally in some sense, subject to inherent psychophysical limitations which constrain the range of his behavior. In the flight control environment, the model is capable of predicting steady-state task performance (e.g., RMS gunsight tracking error), frequency-domain pilot transfer functions (e.g., stick response to a wind gust), frequency-domain pilot *remnant* (e.g., stick jitter), and time-domain dynamic histories (e.g., critical trajectory variables during a piloted *pop-up* maneuver).

A general block diagram of the OCM is given in figure 3.1; a detailed description may be found in Kleinman, et al. (1970). As shown, the model is comprised of the following:

- A set of **vehicle dynamics** which define the basic system states **x** to be controlled by the pilot. In the general flight scenario, these would be vehicle attitude and location in navigation coordinates, vehicle linear and angular velocity, as well as other variables influencing pilot/vehicle response (e.g., SAS dynamics, display delays, etc.).
- A **display interface** which converts system states **x** into displayed variables **y**, seen by the pilot. In the cockpit, these might range from a simple digital indication of airspeed, to an abstracted set of pitch attitude bars, to a highly pictorial volumetric representation of instantaneous weapons envelope. In effect, the display interface converts implicit system states to explicit display variables.

- A **perceptual model** that converts these display variables **y** into noisy and time-delayed *perceived* variables, denoted by $\mathbf{y_p}$. As shown in the diagram, this is accomplished by the formal addition of **observation noise** to the display **y**, followed by a time delay. The noise is included to account for the pilot's visual acuity limits, limitations in his attention-sharing capacity, (which is reflected as imprecision, or noise in perception), and variations in the pilot's response strategy (due to, say, fatigue or idiosyncratic behavior). The noise also accounts for limitations in the man-machine interface, due to such factors as resolution limits, quantization levels, etc. The time delay is included to account for human delays in processing the information available in the cockpit, as well as for inherent delays in the display system itself.

- An **information-processor** or *equalization* block, which converts these perceived variables $\mathbf{y_p}$ to commanded control actions $\mathbf{u_c}$. As shown, this consists of: an optimum (Kalman) estimator, which, in tandem with a predictor, generates a minimum variance estimate of the current system state **x**, and a set of optimal gains which are chosen to ensure that the resulting command $\mathbf{u_c}$ will minimize a pre-defined cost function that expresses the task requirements. In a tactical helicopter situation, the cost function could include a term proportional to the error between target LOS and weapon boresight; in a hovering task, it would include both lateral and vertical deviations from the desired hover point.

- An *equivalent neuromotor model* that converts this command $\mathbf{u_c}$ into a piloted control action **u**. As shown, this is accomplished by the formal addition of *motor noise* to the command $\mathbf{u_c}$, and limited-band-width filtering, which together account for the pilot's neuro-motor bandwidth limitations, and his inability to generate perfectly precise control actions.

A more detailed technical description of the OCM is given in Appendix A.

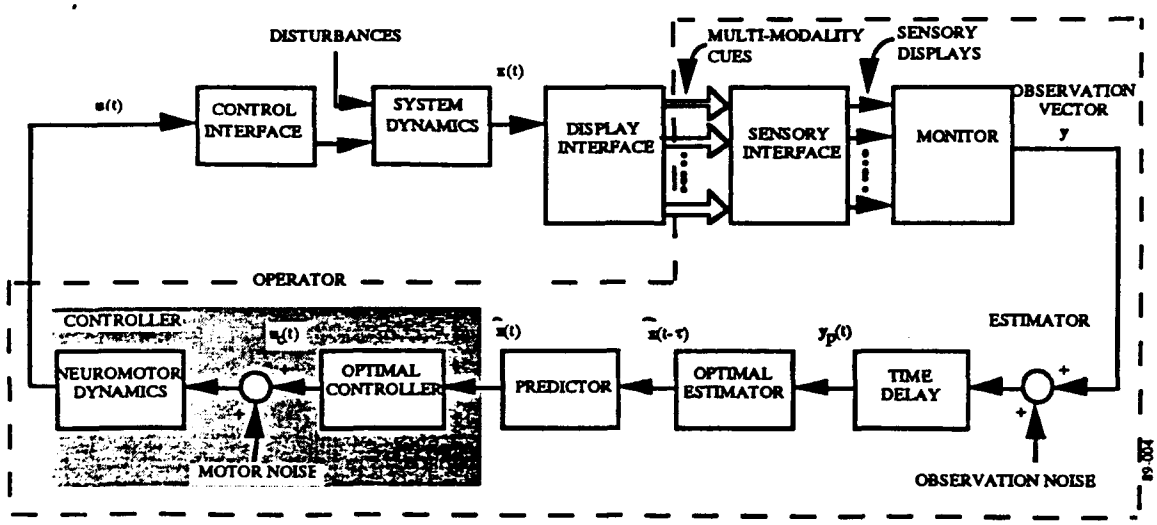


Figure 3.1: Optimal Control Model of Pilot/Vehicle System

The OCM has been validated against experimental data for a variety of tasks, and detailed results may be found in the literature. In one of the earliest validation studies involving closed-loop tracking, it was found that error scores, describing function, and operator randomness were affected by multiple-task requirements all in the manner predicted by the model (Levison, et al. (1969)). Gai and Curry (1976) used the OCM information-processing structure to analyze failure detection in a simple laboratory task, and in an experiment simulating pilot monitoring of an automatic approach. They reported good agreement between predicted and observed detection times for both abstract and realistic situations.

Several studies have used the OCM in realistic workload environments to evaluate pilot behavior. Thus, aircraft display requirements have been investigated via model analysis by Kleinman and Baron (1971), and a model-based instrument display design procedure was developed by Hess (1977). Levison and Baron (1976) analyzed system design effects on flight performance in a terminal configured vehicle (TCV) via the model, and anti-aircraft artillery (AAA) tracking has been similarly modeled, by Kleinman and Perkins (1974). Kleinman and Killingsworth (1974) used the model to predict pilot performance in the flare and touchdown phase of STOL landing. More recently, we have used the model to evaluate the impact of different display designs on terrain-following performance (Zacharias (1985), Brun and Zacharias (1986), Gonsalves and Zacharias (1989)). In short, the OCM has been applied in a number of varied non-laboratory situations, and has provided researchers with a common structure for understanding human performance.

3.2 Artificial Neural Networks for Situation Awareness

A second major enabling technology for CSIM development is artificial neural networks (ANNs), which are particularly suitable for modeling the underlying pattern recognition functions comprising crew situation assessment (SA).

ANNs represent a nonalgorithmic class of information processing for using massively parallel distributed processing architectures (Anderson and Rosenfeld (1988)). Stimulated by the efforts directed at understanding the interconnection of neurons in the human brain allowing the storage, retrieval, and processing of complex data, research over the last 25 years in artificial neural systems has produced solutions to complex problems in visual pattern recognition, combinatorial search, and adaptive signal processing.

There are three main ANN schools of thought. One school approaches the problem from a perception point of view (Rosenblatt (1962)), which leads to the perceptron neural net. The second school uses an associative memory approach (Hopfield (1982)), which results in the Hopfield net. The last school uses a biological approach (Grossberg (1982)), which leads to adaptive resonance theory, ART, neural nets. In his work, Grossberg developed a mathematical model of a biological neuron. The non-linear differential equations that constitute this model are called Grossberg's field equations. From these equations, two simple neural nets can be derived, the instar and the outstar. Both perceptrons and Hopfield nets can be shown to be multiple instar neural nets. A brief comparison of these competing networks are as follows.

- Outstars: An outstar neural network is the minimal network capable of learning and recalling patterns (Hestenes (1983)). Outstars can recall patterns but cannot recognize them.

- Instars: An instar neural network is the minimal network capable of learning and recognizing patterns (Hestenes (1983)). Instars can recognize patterns but cannot recall them.

- Hopfield Nets: A Hopfield net can be considered as multiple instars, which make them useful as classifiers (Hopfield (1982), Hopfield (1984), Hopfield and Tank (1985), Hopfield and Tank (1986), and Koshko (1987)). They are also useful in solving forward error correction, associative memory, and the traveling salesman problems.

- Perceptrons: Perceptrons are basically multiple instars which make them good classifiers; they can recognize but cannot recall (Fukeshima (1988), Widrow and Winter (1988)).

- Adaptive Resonance Theory: ART neural nets are multiple combinations of outstars and instars. ART nets can recognize and recall, and furthermore compare patterns (Carpenter and Grossberg (1988)).

An ANN structure is a network of processing elements (neurons) connected with each other through interconnects (information links). Each processing element can have multiple inputs and only one output. The input/output relationship is described by a first-order differential equation. Specifically, a weighted sum of the nonlinear transformations of the multiple inputs along with a nonlinear transformation of the current neuron's state are the driving functions of this first-order differential equation.

ANNs produce a nearest-neighbor classifier. Since the weighting coefficients change in an unpredictable manner, the global stability of the neural network description is an important consideration. The strongest theoretical result to date is due to Cohen and Grossberg (1983) who have shown that the neural net converges to one of the finite set of equilibrium points corresponding to local minima of the energy function under certain restrictive conditions (e.g., symmetric, positive weighting coefficients).

An advantage of ANNs is that convergence to the answer is independent of the number of local minima in the energy function, thus comparing favorably to otner general search techniques. Although the global stability and convergence results have not been extended to the case for nonsymmetric weighting coefficients, several successful heuristic applications with nonsymmetric weighting coefficients have been reported in Grossberg (1982) and Hecht-Nielsen (1986).

The recent interest in ANNs is due to the availability of a wide range of software packages and the availability of fast, relatively inexpensive computers made possible by advances in VLSI design for realizing neural network structures. Given that the neurons in the human brain process information in milliseconds while outperforming current serial supercomputers with a processing rate in nanoseconds, there is considerable interest in the new generation neurocomputers and computing environments.

In our model, we will incorporate a **perceptron-based backpropagation neural network** which consists of hierarchically connected *layers* of processing units or *nodes*. The nodes of one layer are fully or partially connected to the nodes of the following layer by *paths* which have an associated *weight*. Backpropagation networks typically have an input layer, an output layer, and one or more hidden layers which separate the input and output layers. Figure 3.2 illustrates a generic backpropagation network. The number of nodes in the input layer IL-1 and the number of nodes in the output layer OL-1 are dictated by the given problem. The number

of hidden layers and the number of nodes in the hidden layers are determined by the network developer. The *output* (or *activation*) of node IL-1 travels along path P-1 and becomes the *input* for node HL-1. Each node can have multiple inputs and only one output. The output of a node is determined by the node's *activation function*. The activation function is a nonlinear transformation of the weighted sum of the node's inputs.

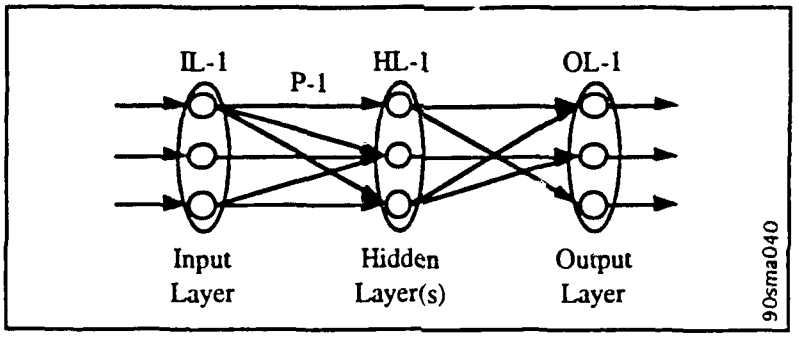


Figure 3.2: Backpropagation Neural Network

In a backpropagation network, each node in the hidden and output layers outputs a weighted summation of its inputs plus a bias after a pass through a threshold nonlinearity (Figure 3.3a). Each node in the input layer outputs its input value unchanged. When continuous output ranges are desired, the *sigmoid activation function* is usually used for the threshold nonlinearity (Figure 3.3b).

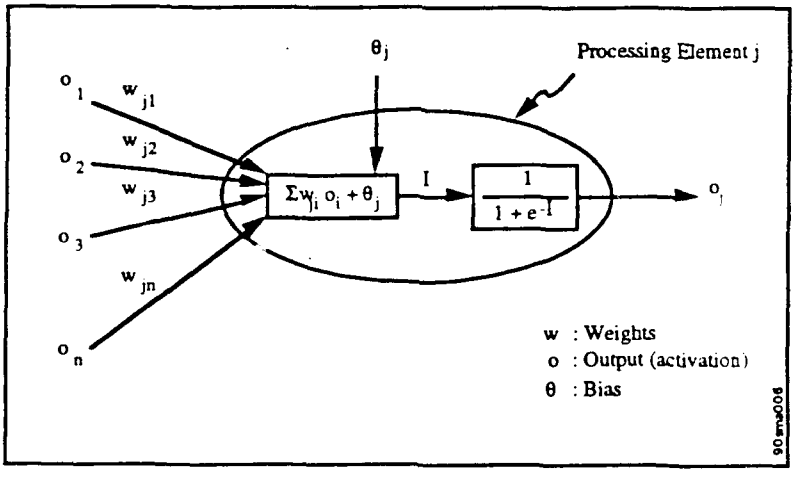


Figure 3.3a: Backpropagation Processing Element

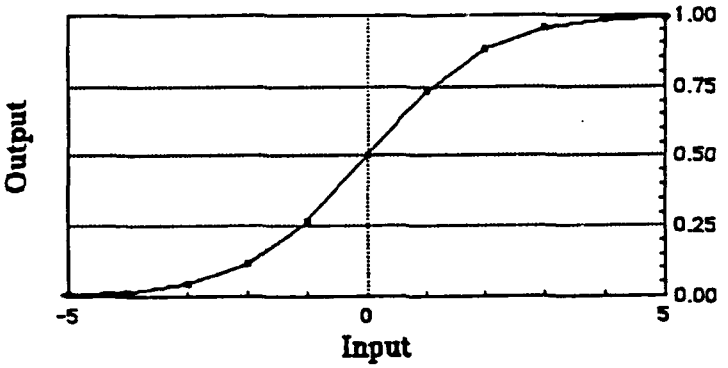


Figure 3.3b: Sigmoid Transfer Function

For backpropagation neural networks to generate the desired output vector for a given input vector, the network must be *trained.* Before training can begin, a set of training data must be created. Training data consists of several sets of input vectors and the corresponding output vectors (usually called IO pairs). Training involves repeatedly presenting the network with the training data. After each IO pair is presented to the network the error between the desired output vector and the actual output vector is calculated and the weights of the network are adjusted to minimize the error according to the Generalized Delta Rule. This *learning* process continues until the difference between the actual output and the desired output is acceptable.

Our particular interest in ANNs lies in their potential for modeling the crew SA functions. ANNs can provide general pattern recognition capabilities in the general state/event space generated by the CSIM information processor (IP), thus supporting the recognition of predefined *situational patterns* occurring over both space and time. A relatively simple example of such a situational pattern is the spatiotemporal pattern generated by, say, an attacking airborne threat, maneuvering through the six spatial dimensions of position and velocity, and the one temporal dimension of time; a more complex pattern could be obtained by combining this maneuver pattern with other hypotheses regarding the threat vehicle characteristics and adversary intent (e.g., vehicle type, weapons complement, adversary ROEs, etc.). Whatever the pattern complexity, we see considerable potential for the use of ANNs for pattern classification, which is effectively situation assessment in this domain of spatiotemporal maneuvers and threat characteristics. Thus, the 7-dimensional spatiotemporal characteristics of the threat maneuver (position, velocity, time) could drive an ANN model of the pilot's *threat maneuver classifier*, which would, in effect, classify the maneuver into one of several potential threat tactics. In conjunction with a knowledge of self-state, and other assessments of threat type and adversary intent, an ANN model could then serve to generate assessments of the overall tactical situation, over the limited domain of the maneuver space, at the time the assessment is being made, and

under the constraints imposed by the limited and imperfect information available to the defending pilot.

Figure 3.4 illustrates the general methodology for ANN modeling of the pilot's threat maneuver classification function. The off-line training process starts with the generation of a database of threat maneuvers where several representative maneuvers are generated for each maneuver class. We then proceed to the definition of an off-line ANN model where we define the external inputs/outputs and interconnect architecture using our a priori knowledge of the constraints of the tactical maneuver envelope. Once the ANN architecture is defined, off-line training proceeds with the repeated presentation of the training data, successive adjustment of the network weights, and eventual convergence of the trained network to an acceptable level of maneuver pattern recognition performance. Following this off-line training stage, the network can then be implemented on-line as part of the overall crew model. As shown in the diagram, the primary input to the trained ANN model are the threat maneuvers but we also envision augmenting the input variable set with additional useful tactical information, specifically, threat vehicle characteristics and inferred adversary intent.

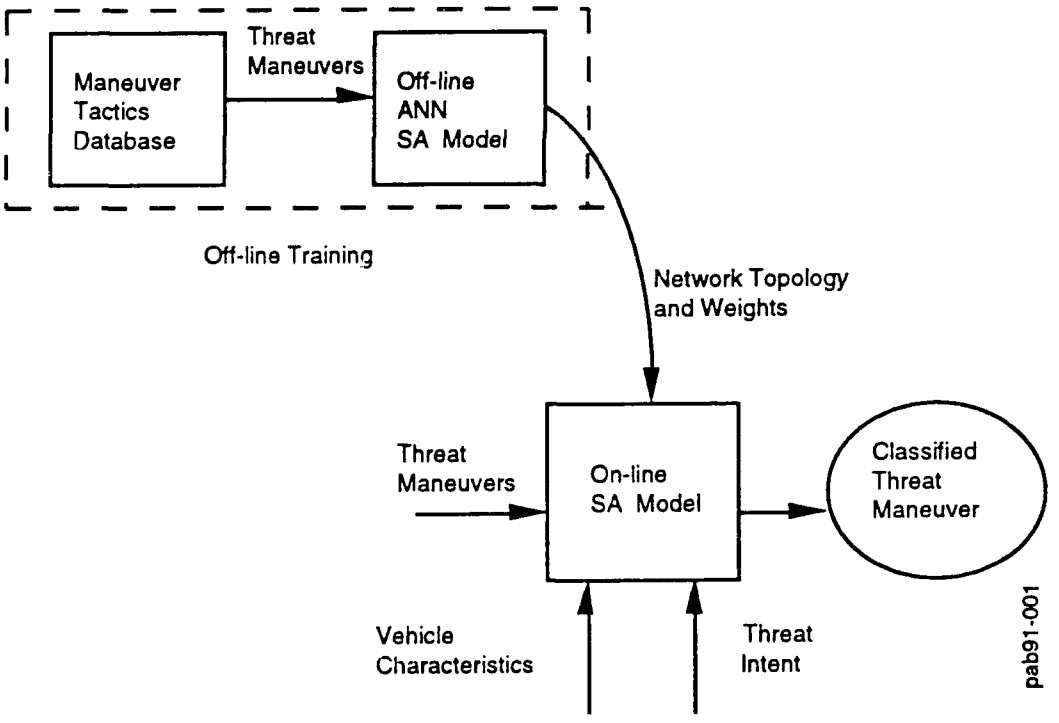


Figure 3.4: ANN Modeling Methodology for Threat Maneuver Classification

Modeling of the pilot's SA function in this manner calls on a number of strengths afforded by ANNs. First, as we have mentioned earlier, ANNs are particularly effective in implementing non-verbal pattern recognition functions which are algorithmically ill-defined and highly perceptual in nature; this is especially the case in the high-dimensional tactical space

defined by the threat trajectory, weapons type, adversary intent, etc., characterizing the engagement. Second, ANNs can provide for considerable data compression with a many-on-few topological mapping from several state/event inputs to a few situational outputs; this type of compression is clearly required if we are to model the non-stimulus driven rational decision-making behavior of the pilot. Third, the node normalization of ANN activation functions can be used to provide classification probabilities in a manner analogous to Bayesian discriminant analysis (Ruck, et al. (1990), Wan (1990)). Since ANN classification in the SA context need not be an all-or-nothing affair, a potential exists for modeling human hypothesis SA ranking according to the confidence in that hypothesis held by the pilot. Finally, an ANN model has the potential for modeling on-line unsupervised learning of tactically relevant patterns, and the sharpening of tactical skills that comes with repeated successful engagements.

3.3 Expert Systems for Decision-Making and Procedure Selection

A third enabling technology for CSIM development is expert systems (ESs), which can provide a natural framework for modeling the crew's decision-making (DM) and procedure selection functions.

The recent success of ES technology (Stefik, et al. (1982)), in diagnosis, monitoring, prediction, planning, tracking, and design problems in certain application domains has initiated similar efforts in other areas. These early successful expert system applications include SOPHIE in computer assisted instruction (Brown, et al. (1974)), MYCIN in medical diagnosis (Shortliffe (1976)), PROSPECTOR in oil exploration (Duda, et al. (1978)), and DENDRAL in biology (Buchanan and Feigenbaum (1978)). Recent military applications of expert systems include ADEPT for battlefield situation assessment analysis (Taylor, et al. (1984)), EXPERT NAVIGATOR for monitoring aircraft navigation sensors (Pisano and Jones (1984)), and ACEM for modeling air combat (Mitchell (1989)). In addition, a number of avionics applications are underway, including the USAF-sponsored Pilot's Associates development system (Lizza (1989), Corrigan and Keller (1989)), the NASA-sponsored effort for Situation Assessment and Response Aiding (SARA, Hudlicka (1989)), and the Army-Sponsored A^3I program (Corker (1986)).

An expert system is a computer program that can perform a task normally requiring the reasoning ability of a human expert. ESs are highly specialized according to their application domains. Although any program solving a particular problem may be considered to exhibit expert behavior, ESs are differentiated from other programs according to the manner in which the domain specific knowledge is structured, represented, and processed to produce solutions. In particular, ES programs partition their knowledge into the following three blocks: Data Base, Rule Base, and Inference Engine. ESs use symbolic and numeric reasoning in applying the rules

in the Rule Base to the facts in the Data Base to reach conclusions according to the construct of reasoning specified by the Inference Engine.

3.3.1 Knowledge Representation

There are two basic types of knowledge that can be incorporated into ESs: **declarative knowledge** and **procedural knowledge**. The kind of knowledge describing the relationships among objects is called declarative knowledge. The kind of knowledge prescribing the sequences of actions that can be applied to this declarative knowledge is called procedural knowledge. In ESs, procedural knowledge is represented by production rules whereas declarative knowledge is represented by frames and semantic networks, in addition to production rules.

Rules are expressed as IF-THEN statements. When the IF portion of a rule is satisfied by the facts, the rule is fired by executing the statements specified by the THEN portion. Typically, the production rules deal with uncertainty through the use of certainty factors, probability or fuzzy logic. Semantic nets are network representations of declarative knowledge. A semantic net consists of a collection of nodes - representing arbitrary objects - connected by arcs describing the relations between nodes. One of the most important characteristics of a semantic net is the capability of building inheritance hierarchies. Using arcs representing relations such as ISA and HAS-PART, objects in the net can inherit properties from other objects higher up in the net. A frame is a knowledge representation about a prototypical instance (Fikes and Kohler (1985)). Frames are organized as semantic nets where the topmost nodes represent general concepts whereas the lower nodes represent more specific instances of these concepts. In a frame based system, the concept at each node is defined by its attributes (slots) and attribute values. Each slot can also contain procedures which are executed when the values of the attribute change.

While expert systems have traditionally been built using collections of rules based on empirical associations, interest has grown recently in **knowledge-based expert systems** which perform reasoning from representations of structure and function knowledge. For instance, an ES for digital electronic systems trouble shooting is developed by using a structural and behavioral description of digital circuits (Davis, et al. (1982), Davis (1983), Davis (1984), Davis (1987)). Qualitative process theory is another approach allowing the representation of causal behavior based on a qualitative representation of numerical knowledge using predicate calculus (Forbus (1982), Forbus (1984) Forbus (1987) Forbus (1988)).

3.3.2 Inference Strategies

The inference control strategy is the process of directing the symbolic search associated with the underlying type of knowledge represented in an expert system: antecedents of IF-THEN rules, nodes of a semantic net, or a collection of frames. In practical ES applications, the blind

search is an unacceptable approach due to the associated combinatorial explosion. Search techniques can be basically grouped into three: breadth-first, depth-first and heuristic. The breadth-first search exhausts all nodes at a given level before going to the next level. In contrast, the depth-first exhausts all nodes in a given branch before backtracking to another branch at a given level. Heuristic search incorporates general and domain-specific rules of thumb to constrain a search.

Expert systems employ basically two types of reasoning strategies based on the search techniques above: forward chaining and backward chaining. In forward chaining, starting from what is initially known, a chain of inferences is made until a solution is reached or determined to be unattainable. For instance, in rule based systems, the inference engine matches the left-hand side of rules against the known facts, and executes the right-hand side of the rule that is activated. In contrast, backward-chaining systems start with a goal and searches for evidence to support that goal. Pure forward chaining is appropriate when there are multiple goal states and a single initial state whereas backward chaining is more appropriate when there is a single goal state and multiple initial facts. Many expert systems use both forward and backward chaining.

The potential utility of an ES implementation of the crew's decision-making DM and procedure execution activities lies in four areas. First, there is a natural mapping of the ES architecture to the crew DM functions: the ES data base represents the assessed situation (S), the ES rule base reflects the crew's procedure set, and the ES inference engine models the crew's DM algorithm(s). Second, an ES implementation can readily incorporate heuristics employed by the crew to *short-cut* extensive decision/procedure sequences, in accord with the stimulus-driven activity path postulated by Rasmussen (1983). Third, ESs can implement dynamic updating of their knowledge base, to accou[illegible] for short-term trends peculiar to the mission phase or task. Finally, ESs can serve as a knowledge repository for implementation of the *upstream* crew functions of information processing (IP) and situation assessment (SA). In particular, ESs can maintain a knowledge base of IP techniques and modes incorporated in the modern estimation (ME) formulation used to model the crew; likewise, ESs can characterize SA strategies and procedures represented in an ANN formulation of the SA function, via a specification of network topologies and weighting schemes used for different SA objectives during a mission.

3.4 Hybrid System Development Tool - *NueX*

The general hybrid development environment for ANN/ES integration is illustrated in figure 3.5. This hybrid environment, referred to as *NueX*, is being developed by Charles River Analytics. Within *NueX*, expert knowledge is handled by an expert system shell, and the neural network descriptions are stored in external data structures; a HyperCard interface serves as the

communication link between them, in addition to the application development shell. In particular, the input and output nodes of a neural network are defined within the knowledge base, while the complete network architecture remains external. In this manner, the neural network nodes are related to application specific objects, thereby inheriting physical properties, while the internal ANN structure is efficient and modular. This modularity offers the ability to incorporate ANNs which operate on parallel rather than serial processing. In addition, rules can be created to perform ANN development and training procedures, therefore providing a strong foundation for automation.

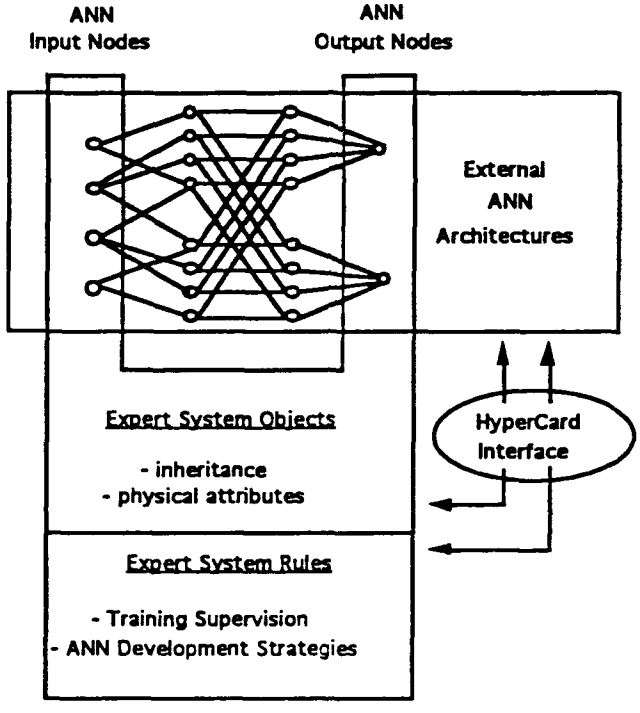


Figure 3.5: *NueX* Hybrid Environment

NueX provides an application development system incorporating both neural networks and expert systems within the functionality of the Apple Macintosh HyperCard environment. The *NueX* development environment is illustrated in figure 3.6. The environment is divided into three main sections: Neural Networks, Expert Systems, and Hybrid. Typically, the development process begins with the ANN and knowledge base development. After developing the ANN architectures and setting up the KB rules, the links between ANN input/output nodes and specific KB objects/properties are defined. ANN training is accomplished either interactively or automatically by using the expert neural training which performs the learning supervision. ANNs and KBs are then tested within their respective sections, and design iterations are performed as needed. Within the Hybrid section, the specific application user interface is created by taking

advantage of HyperCard's object-oriented visual programming language in addition to predefined graphical and data I/O routines. We now describe the specific elements within *NueX*.

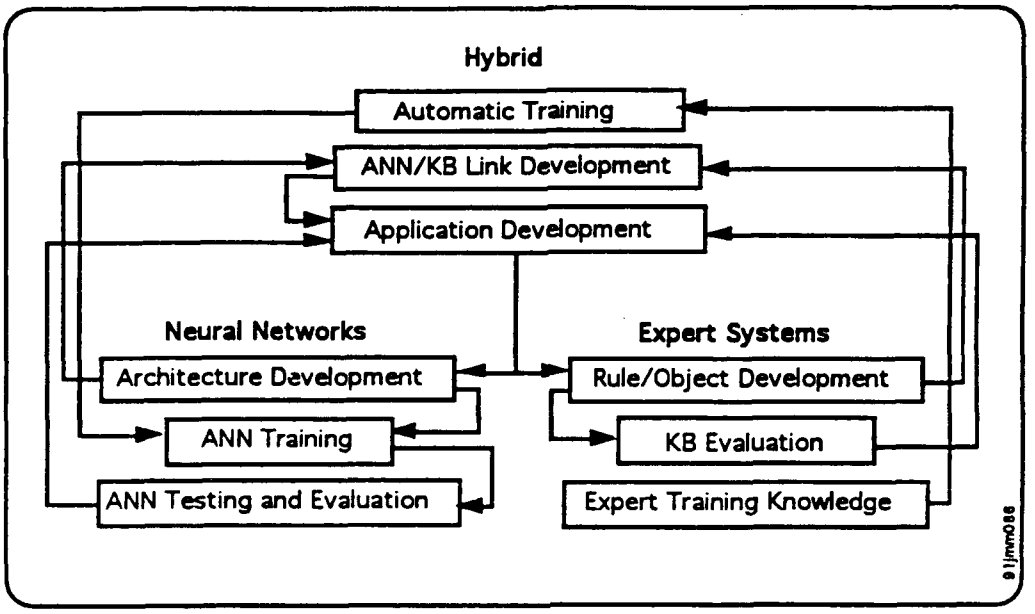


Figure 3.6: *NueX* Hybrid Development Environment

3.4.1 Neural Network

Architecture Development

The ANN Architecture Development section of the hybrid *NueX* environment is illustrated in figure 3.7. Shown graphically is a target classification network we used for a Navy sponsored multiple target recognition (MTR) study (Gonsalves and Caglayan (1992)). It has two input nodes, ten hidden nodes, and a single output node which is used for target classification. The palette to the left of the development window contains the tools necessary for specifying the ANN interconnect structure. Using this palette, the user can move a node (the currently selected option in figure 3.7), create and erase nodes and paths, and display information separately for each path and node.

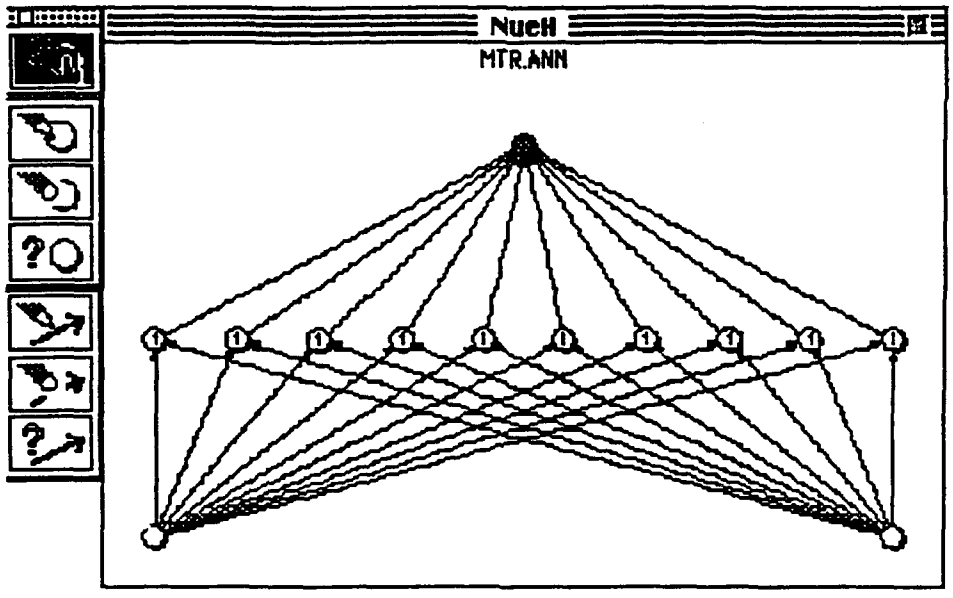


Figure 3.7: ANN Architecture Development

Neural Network Training and Evaluation

In the interactive training option, the user is required to monitor and control all aspects of the training process. The following functions are available:

Deviate Weights	- *deviate weights randomly or by user-specified amount*
Reset Weights	- reset all weights randomly with specified range
Cycle Order	- user-defined layer propagation cycle order
Graph Freq.	- specify graphing frequency (cycles/graph update)
Learning Cycles	- specify number of learning cycle iterations
Randomize IOPs	- automatically randomize training patterns
Start/Stop Recording	- save RMS error plotting data to a text file
Use Biases	- set the network's bias output between 0 and 1

Figure 3.8 shows the graphical interface for training and evaluating neural networks. The current RMS training error is plotted using a logarithmic scale; this serves to visually amplify the RMS signal as the network approaches convergence. The current RMS error is continually updated graphically and within the text fields as the learning progresses. The ANN's current learning rate, momentum, and bias output are also displayed within text fields.

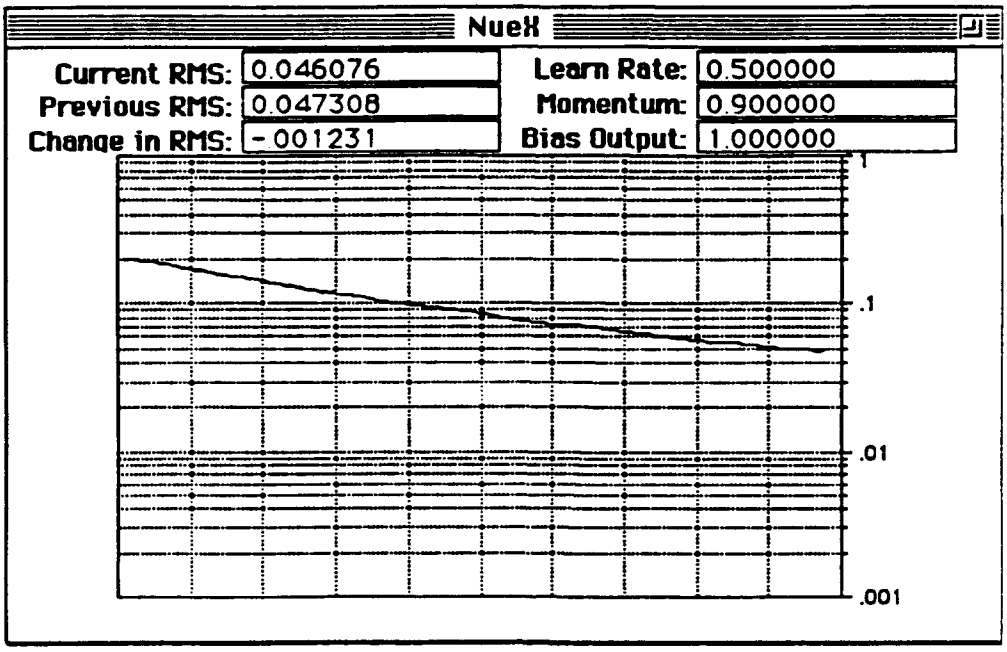


Figure 3.8: ANN Training and Evaluation

3.4.2 Knowledge Base Development and Evaluation

NueX provides the functionality to design, develop, and evaluate Expert System knowledge bases. This functionality is provided with a Knowledge Base editor for developing rule base, a Knowledge Base monitor to test and evaluate the inference performance of the developed rule base. Shown in figure 3.9 is the KB editor interface used to create rules. The rule shown here is from a target classifier knowledge base used in our prototype hybrid Multi-Target Recognition system developed for the US Navy. The elements of the editor that pertain to rule generation include, rule name, rule firing priority, the IF or conditional part, and the THEN part or the facts that get asserted when the rule is fired. The rule name is the symbolic name the user can assign to the specific rules to ease in the knowledge base development and evaluation process. The rule firing priority is an integer value greater than or equal to zero. The higher the integer, the earlier the rule will fire. The conditional part of the rule is given in the text window region under the IF in figure 3.9. Logical expressions are input here by the user. The example shown here looks at current values of range, range rate, and cosine of Target Aspect Angle (TAA) to see if certain conditions are met. The user can select whether to employ the logical AND or the logical OR when evaluating the conditional. This done by checking the boxes next to AND OR, respectively, as shown in figure 3.9.

The facts that get asserted when the rule is fired are in text window below the THEN. Elements of this text window are scripted using a basic knowledge of HyperCard's scripting language HyperTalk. For instance, in the rule shown if the IF part is true the fact asserted is that

KB_output is "hostile". Rules can be reset at the end of an inference cycle by checking the box next to RESET. *NueX* also affords other capabilities for knowledge base development including, definition of global variables, use of default values for global variables, cutting and pasting rules, and capabilities for accessing and calling ANNs and other knowledge bases.

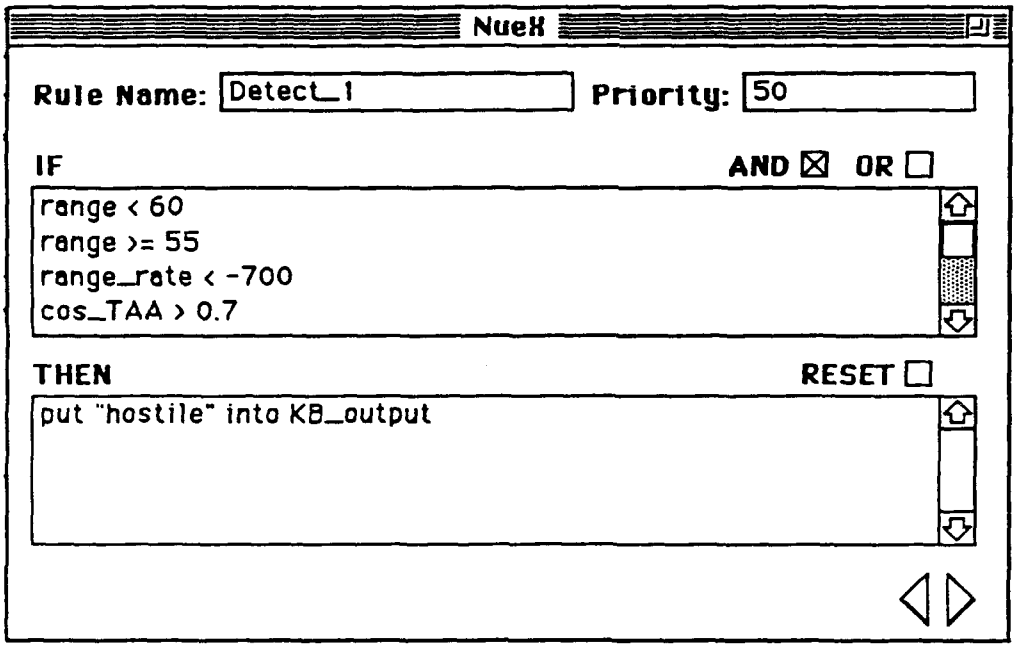


Figure 3.9: Knowledge Base Development

Knowledge bases can be tested and evaluated using the KB monitor. The user can test and evaluate the knowledge by either running through the whole inferencing process or stepping through one rule at a time. While running the knowledge base, the current invoked rule is shown highlighted and any rules fired are listed in order of firing in the column to the left of the rule names. During the knowledge base run through, the values for the global variables are updated on the screen. The step through is similar except the inferencing stops when a rule is fired. The user can check the global variables before stepping through until the next rule fires. In a such a manner, a knowledge base can be debugged, tested, and evaluated.

4. MODEL-BASED ANALYSIS OF SITUATION DISPLAY

As part of our feasibility assessment of model-based display analysis, we conducted a limited-scope evaluation of alternative display options for a selected flight task, using a subset of the crew/system integration model (CSIM) described in the previous chapters. This chapter describes the modeling effort itself, to better define the flight task and to provide an indication of the steps required for model-based analysis. The next chapter follows up with an evaluation of the model-generated results, and an assessment of display options based on performance awareness metrics generated in the CSIM environment.

This chapter is organized into three sections. Section 4.1 describes the overall system implementation. Section 4.2 details the external and vehicle modules, and subsystem modules. Section 4.3 then concludes with a description of the crew-centered portion of the modeling effort.

4.1 Overall System Implementation

The model implementation effort conducted here focused on implementing a limited scope version of the CSIM model to demonstrate feasibility of the overall approach. Here we review some aspects of this limited scope version and concentrate on the representation of three major functional blocks: 1) the external world modules, representing targets, threats, weather, and terrain; 2) the ownship modules representing the vehicle dynamics subsystems and displays; and 3) the crewmember modules which act to process the displayed information and effect the appropriate procedures. Figure 4.1 provides an overall data flow diagram of the CSIM implementation.

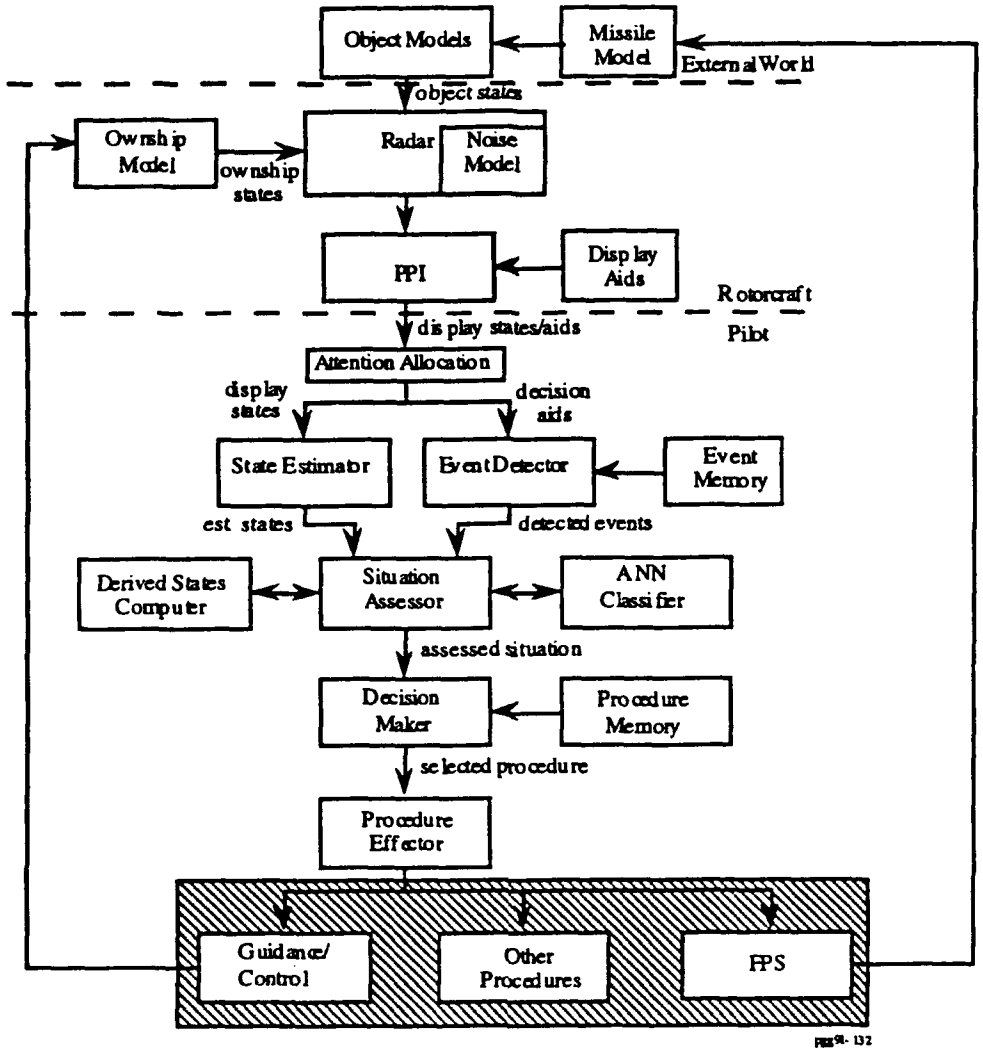


Figure 4.1: CSIM Implementation: Overall Data Flow

The external world modules represent the object models, specifically, the threats and friendlies in the environment, and the missile model. Simplified point mass equations are used for all models in the environment. Additional details are provided in the next section regarding specific implementation.

The ownship modules represent the ownship model dynamics, its subsystems, and the displays and display aids that provide the crew information during the mission. In this particular modeling effort, the ownship model dynamics are again represented as maneuvering point mass equations. The vehicle display is limited to the radar subsystem and associated PPI display. In addition, specific consideration is given to display aids to aid the pilot in situation assessment and decision making. Additional details are found in the following section.

The pilot module represents the crewmember's sensory and perceptual processing of the displays, down stream information processing of continuous states and discrete events, situation assessment, decision making, and procedure execution. The module illustrated here is a specific instantiation of the CSIM architecture presented in chapter 2, specialized to the tactical rotorcraft scenario.

We have simplified the sensory channel modeling by restricting the sensory inputs to those visual cues available through the Tactical Situation Display (TSD). Attention allocation amongst those cues is chosen so as to maximize state estimation accuracy and situation assessment reliability. No other cues have been modeled for this effort, although it is clear that additional information is available through out-the-window displays and non-visual cueing.

The state estimation and event detection modules model the crew's processing of available display states and decision aids. The state estimator serves to generate a best estimate of the ownship relative to threat and friendly states, including position, velocity, and acceleration. The event detector detects decision aids provided by the enhanced TSD, and matches them with an event memory to assert the presence of a detected event.

Both the estimated states and the detected events feed the situation assessor. The situation assessor gathers and combines the evidence set obtained to generate an assessed situation $\hat{S}_i$ that represents the crewmember's assessment of the current and near-term tactical situation. As described below, three components of the tactical situation are considered in this formulation: 1) threat versus friendly assessment, or Identification, Friend or Foe (IFF); 2) high versus low priority assessment, or Threat Prioritization (TP); and 3) within or outside missile firing range, or Fire Point Selection (FPS). For each situational event, a situational occurrence probability is computed. As shown in the diagram, a derived states' computer feeds the situation assessor, and classification is accomplished via an ANN classifier.

Once an assessed situation is generated, it is fed to a decision maker, which incorporates rule-based decision making regarding threat/friendly classification, prioritization, and fire-point selection. Once an action decision has been made, a procedure effector performs one of three functions: a) guidance; b) control; and c) missile firing.

We now describe these component modules and functions in greater detail, beginning in section 4.2 with the ownship description, and proceeding in section 4.3 with the crewmember description.

4.2 External and Vehicle Modules

4.2.1 Object and Ownship Modules

The target and ownship modules simulate multiple hostile and friendly targets as well as the ownship platform. The modules generate time histories of target and ownship position, velocity, and acceleration. The target module currently supports two modes of trajectory generation operation: 1) trajectories employing a line-of-sight (LOS) intercept; and 2) trajectories using zero acceleration/constant velocity. In general, the object module implements the following vector differential equations:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t)$$
$$\dot{\mathbf{v}}(t) = \mathbf{a}(t) \qquad (4.1)$$

where $\mathbf{r}(t)$ is the target's position at time t, $\mathbf{v}(t)$ its velocity, and $\mathbf{a}(t)$ its acceleration. The acceleration is defined for two different guidance schemes in the following manner. For constant velocity it is given by:

$$\mathbf{a}(t)=0 \qquad (4.2a)$$

For the LOS intercept it is:

$$\mathbf{a}(t) = K \text{ sign } [\omega . (\rho \times \mathbf{v})] \dot{\rho} (\mathbf{u}_\rho \times \omega) \qquad (4.2b)$$

where ρ is the relative position vector, $\dot{\rho}$ is range rate, $\mathbf{u}_\rho$ the unit line-of-sight (LOS) vector, K is a constant, and ω is the angular velocity vector of the LOS.

The dynamics for the ownship model are given by the following:

$$\dot{x} = V \cos(\gamma) \cos(\psi)$$

$$\dot{y} = V \cos(\gamma) \sin(\psi)$$

$$\dot{z} = -V \sin(\gamma)$$

$$\dot{V} = g(n_x - \sin(\gamma))$$

$$\dot{\gamma} = \frac{g}{V}(n_z \cos(\phi) - \cos(\gamma))$$

$$\dot{\psi} = \frac{g\, n_z \sin(\phi)}{V \cos(\gamma)} \qquad (4.3)$$

where x, y, z are ownship position coordinates, v is total velocity, γ is flight path, and ψ is ownship heading. The dynamics are driven by the control triplet (n_x, n_z, ϕ) where n_x is the longitudinal load factor, n_z is the normal load factor, and ϕ is the bank angle.

4.2.2 Radar and PPI Modules

Figure 4.2 presents the block diagram of the radar and TSD models. The radar sensor path is driven by the target and ownship states. Their difference yields an ownship relative set of position and velocity states (ρ and $\dot{\rho}$). These relative states are then transformed from Cartesian into spherical coordinates. The resulting six dimensional spherical states (range (ρ), elevation (θ), azimuth (Ψ), range rate ($\dot{\rho}$), elevation rate ($\dot{\theta}$), and azimuth rate ($\dot{\Psi}$)) are then corrupted with noise to simulate radar sensor measurements. The range and range rate signals are corrupted via multiplicative noise, i.e.,

$$\begin{aligned}\tilde{\rho}(t) &= \rho(t) + k\,\rho(t)\,w_\rho(t)\\ \tilde{\dot{\rho}}(t) &= \dot{\rho}(t) + k\,\dot{\rho}(t)\,w_{\dot{\rho}}(t)\end{aligned} \tag{4.4}$$

where the ~ denotes the noisy measurement, k is the multiplicative factor, and w are the white noise sources. The angles and angular rates are corrupted with additive noise in the following manner:

$$\begin{aligned}\tilde{\theta}(t) &= \theta(t) + w_\theta(t)\\ \tilde{\psi}(t) &= \psi(t) + w_\psi(t)\\ \tilde{\dot{\theta}}(t) &= \dot{\theta}(t) + w_{\dot{\theta}}(t)\\ \tilde{\dot{\psi}}(t) &= \dot{\psi}(t) + w_{\dot{\psi}}(t)\end{aligned} \tag{4.5}$$

with again the ~ denotes the noisy measurements and w are the white noise sources.

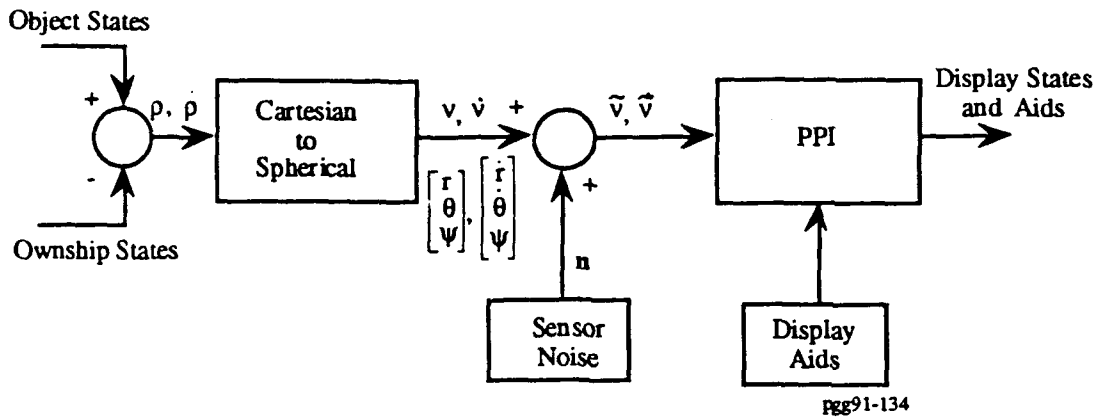


Figure 4.2: CSIM Implementation — Radar/TSD Modules

As shown in figure 4.2, the TSD module transforms the radar measurements into displayed states. Also, any display aids to be included in the TSD are made available to the pilot. The TSD used for this study is based on the situation display developed at the Army's Human Engineering Laboratory (HEL) and documented in Dominessy, et al. (1991) The TSD was developed to provide Army helicopter pilots tactical information to help them locate targets while simultaneously flying the vehicle. Figure 4.3 shows the TSD split screen which provides for:

1) An upper navigation display for airspeed, altitude, rate of climb, heading, and engine torque data; and

2) A lower situation display which indicates friendlies (circle), hostiles (diamonds), and unknowns (U's), in an ownship-centered, heading up, top-down view of the tactical area.

Our focus is on the bottom display, which differentiates rotorcraft from fixed-wing aircraft by use of an overbar, and which provides a heading vector and track number for each object displayed. The large circle is a range indicator.

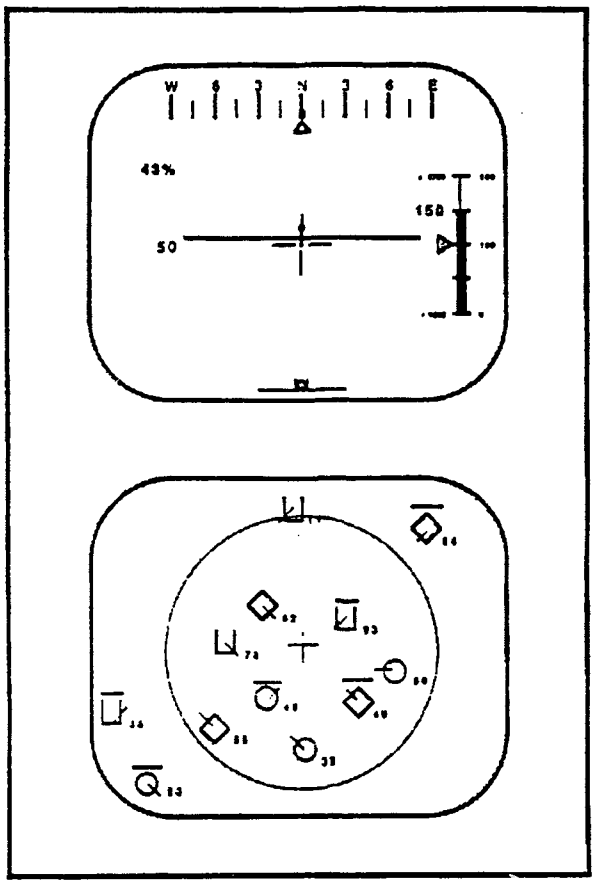


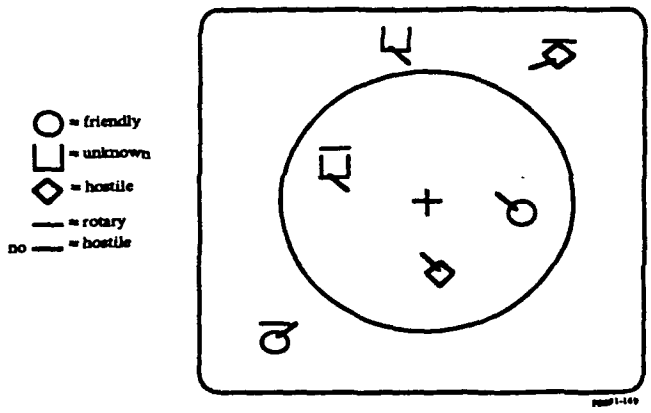
Figure 4.3: HEL Tactical Situation Display

Our implementation of the TSD display is shown in figure 4.4. The display consists of:

- crosshair at the center of the display presenting ownship position
- range ring at ranges of 5 km and 10 km
- icons representing object presence, threat status (friendly, unknown, or hostile) and threat type (rotorcraft or fixed wing).

The information set derived from the display consists of continuous and discrete information. Continuous information includes two-dimensional object location, in terms of relative range and heading. Derivative information (i.e., range rate and heading rate) is made available via the velocity vector. Discrete information consists of IFF object tagging via object icons, threat prioritization via icon color, and fire-point selection (FPS) via flashing icon.

- **Display**



- **Display Attributes**
  - Inside-Out, Heading Up, Relative **r,v**
- **Information Set**
  - Continuous Variables
    - Object Location (r,ψ)
    - Object Velocity (rdot, ψdot) - via vector
  - Discrete Variables (⊔, ○, ◇)
    - IFF object tagging
    - Prioritization (by color)
    - Fire Point selection (FPS) (flashing)

Figure 4.4: Situation Display

4.3 Pilot Modules

To complete the model specification, we need to specify the pilot's: a) visual perception model; b) state estimator; c) situation assessor; d) decision maker; and e) procedure effector.

4.3.1 Visual Perception Model

To specify the visual perception model , we need to specify: a) the display variable set; b) the associated noise variable levels; and c) the attention allocation among the display elements.

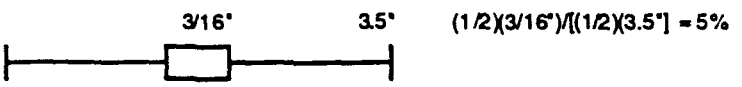
From the previous section, the continuous display variable set, $\mathbf{y}_i$, is composed of the following:

$$\mathbf{y}_i = [\rho_i \; \psi \; \dot{\rho}_i \; \dot{\psi}_i]^T$$

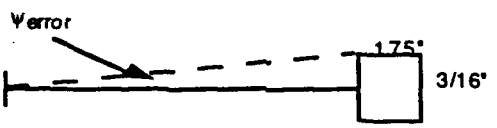
where ρ is range, ψ is heading, $\dot{\rho}$ is range rate, and $\dot{\psi}$ is heading rate for the ith object on the display. For N such objects, we then have a 4N-dimensional display vector. We now need to specify the associated noise levels for each of these continuous display variables. For the discrete variables, IFF object tagging, threat prioritization, and FPS information, we assume no noise (i.e., perfect information).

Figure 4.5 summarizes the display analysis for a single tagged object on the TSD. As described in the HEL Counter-Air Situation Awareness Display Study (1991), a nominal 28 in. viewing distance and a visual acuity range of 10 to 25 arc-min calls for a recommended icon size of 3/16 in. If the display area is 7 in. x 7 in., then the approximate range resolution (in %) will be one-half the symbol size divided by half the display radius (for an average radial error) to yield an approximate 5% error as shown. We assume a proportional scaling of range errors at this 5% level, over the full domain of the display. In addition, we assume a range rate noise level of 5% as well, to equal the range noise level; this selection is based on previous pilot model-based display analyses (e.g., Kleinman, et al. (1971), Zacharias (1985), Gonsalves, et al. (1991)). The angular (Ψ) or heading resolution is computed as one-half the symbol size, viewed at the maximum display radius, to yield a 3.1 degree error as shown. This is rounded off and approximated as 5 degrees. In addition, we assume an angular rate noise level of 0.5 deg/s, which reflects the relatively good angular rate information provided by the object velocity vector tags. In table 4.1 we list the associated noise statistics for each of the continuous display variables.

- **Display Symbol Size (HEL Symbol Size Study)**
  - Acuity: 23 arc min
  - Viewing Distance: 28"
  - Symbol Size: 3/16"
- **Resulting Radial Noise Levels**
  - 7" x 7" Display Size
  - Average Noise Level:



  - Choose Range Rate Noise Proportional to Range Noise
- **Resulting Angular Noise Levels**
  - Average Noise Level



  - ψ error = 3.1° (approximate as 5%)
- **No Noise on Discretes**

Figure 4.5: Display Analysis for TSD

Table 4.1: Display Variable Noise Statistics

Display Variable	Symbol	Noise Level
Range	ρ	5%
Range Rate	$\dot{\rho}$	5%
Heading	ψ	5°
Heading Rate	$\dot{\psi}$	0.5°/s

Attention allocation of the display elements is assumed to be divided equally among the number of objects being tracked on the display. For N objects, the attention on the ith object, f_i, is then:

$$f_i = \frac{1}{N} \qquad (4..6)$$

The associated noise level for each of the four display elements (ρ , $\dot{\rho}$, ψ, $\dot{\psi}$) for the ith object are then scaled as inversely proportionally to the allocated attention Gonsalves and Zacharias (1989), in accordance with:

$$\sigma_i = \frac{1}{f_i}\sigma \tag{4.7}$$

where σ is the nominal noise level from table 4.1 and σ_i is the new scaled noise level.

4.3.2 State Estimator and Discrete Event Detector

Figure 4.6 shows a block diagram of the state estimator. The estimator takes in the display information set V composed of range, range rate, heading, and heading rate for each tracked object. These spherical measurements are in turn converted to Cartesian coordinates and passed through an Extended Kalman Filter to produce the pilot's estimates of target states.

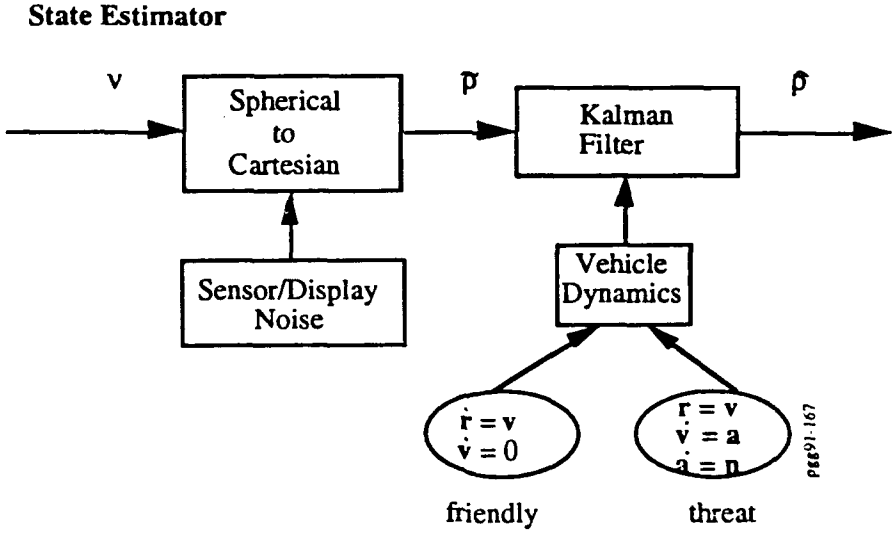


Figure 4.6: State Estimator Block Diagram

The dynamic state estimator is based on the extended Kalman Filter algorithm. As general background, consider the linear discrete-time system described by

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{x}_k + \mathbf{E}\mathbf{w}_k \tag{4.8}$$

where **x** is a vector of system states, **u** the controls, and where **w** the process noise is a vector of random sequences with zero mean, no time correlation (i.e. *white* noise), and with covariance matrix **Q**. At each time step k, a measurement $\mathbf{z}_k$ is made, which is related to the state $\mathbf{x}_k$ by

$$\mathbf{z}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k \tag{4.9}$$

where **v** is *white* measurement noise with zero mean and a covariance matrix **R**.

Given the noisy measurement $\mathbf{z_k}$, the extended Kalman Filter (EKF) algorithm (Bryson (1975)) computes the least mean sequence estimate of the state via the following equations

$$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_{k+1} + \mathbf{K}_{k+1}\left[\mathbf{z}_{k+1} - \mathbf{C}\bar{\mathbf{x}}_{k+1}\right] \tag{4.10}$$

$$\mathbf{P}_{k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\,\mathbf{C})\;\mathbf{M}_{k+1} \tag{4.11}$$

and the *a priori* state estimate and error covariance matrix propagated as

$$\bar{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k \tag{4.12}$$

$$\bar{\mathbf{M}}_{k+1} = \mathbf{A}\mathbf{P}_k\mathbf{A}^T + \mathbf{E}\mathbf{Q}\mathbf{E}^T \tag{4.13}$$

with the filter gain $\mathbf{K_k}$ given by

$$\mathbf{K}_{k+1} = \mathbf{M}_{k+1}\,\mathbf{C}^T\left(\mathbf{C}\mathbf{M}_{k+1}\,\mathbf{C}^T + \mathbf{R}_{k+1}\right) \tag{4.14}$$

We now apply the extended Kalman Filter formulation to the estimation problem at hand. We use a relative coordinate system in which only the target is accelerating. The system dynamics model is then given by

$$\begin{aligned} \dot{\mathbf{v}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{a} \\ \dot{\mathbf{a}} &= \mathbf{n} \end{aligned} \tag{4.15}$$

for a hostile object and given by

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{o} \end{aligned} \tag{4.16}$$

for a friendly where **r** and **v** are the 2-dimensional relative position and velocity vectors, **a** is relative acceleration, and **n** is a white noise process. For our implementation we make the assumption that the pilot has the correct internal estimator system dynamics model, i.e., non-maneuvering for a friendly object and maneuvering for a hostile object.

The discrete event detector for this implementation assumes a nominal pass through of the displayed discretes (IFF, priority, and FPS), with zero detection times and no errors. Therefore, as soon as a discrete event (e.g., IFF tagging) occurs or is displayed on the TSD, the pilot is assumed to process that information instantaneously and without error.

4.3.3 Situation Assessor

The **situation assessor** module gathers and combines the evidence set obtained from the state estimator and discrete event detector to generate an assessed situation S_i that represents the pilot's *picture* of the current and near-term tactical situation. Once the situation is assessed, it is assumed the crewmember *immediately* starts an action procedure that is triggered by the situation. The rules in the situation assessor use combinations of facts in the fact base to reach general conclusions about the current situation. Some of these rules represent *rules of thumb* that a pilot would follow given all the available information about the situation. The facts generated by the situation assessor along with those generated by the event detector combine to describe the assessed situation.

For our study, we assumed that the pilot's assessed situation was based on three situational components defined for each tactical object i:

e_i = ith object is a threat (4.17a)

f_i = ith object is high priority

g_i = ith object is within missile firing range (with FPS)

we can then define a situation vector for the ith object given by:

$$S_i = \begin{bmatrix} e_i(t) \\ f_i(t) \\ g_i(t) \end{bmatrix} \quad (4.17b)$$

where we have introduced the temporal dependence to emphasize the changing situation over the course of the engagement. For each of these situational components, we can define an *assessed* situational component, defined by the crewmembers probability assessment that each component is true. Specifically, we can define three corresponding occurrence probabilities:

$P\{e_i\}$= probability that ith object is a threat (4.18a)

$P\{f_i\}$= probability that ith object is high priority

$P\{g_i\}$= probability that ith object is within FPS

We can then define an assessed situation vector corresponding to the actual defined in (4.17b), via:

$$\hat{S}_i = \begin{bmatrix} P\{e_i(t)\} \\ P\{g_i(t)\} \\ P\{f_i(t)\} \end{bmatrix} \tag{4.18b}$$

We now describe how these occurrence probabilities of (4.18) are generated.

We model threat/friendly assessment as primarily one of situational pattern recognition. In effect, there exists the need to classify the spatiotemporal pattern generated by an unknown (threat/friendly) in the tactical airspace, characterized by observed changes in object *state* during possibly threatening maneuvers. We model this overall process as shown in figure 4.7. Here, the **state estimator** just described generates dynamic running estimates of the unknown object's position, velocity, and acceleration, based on the information available to the ownship pilot. Note, of course, that these estimates are imperfect because of limitations in both the pilot's perceptual capabilities and the TSD limitations. These unknown object states are then processed by a **maneuver parameter generator** (implemented via the derived states computer) which attempts to identify the tactics (if any) being used by the object of concern. For our study we evaluated the *truth value* of the tactical hypotheses that the object was an attacking threat, following a line-of-sight (LOS) intercept guidance trajectory to intercept the ownship. This evaluation is accomplished by noting that a LOS intercept guidance tactic calls for the threat to accelerate toward the ownship in accordance with the guidance law given earlier in (4.2b).

Figure 4.8 shows the maneuver parameter generator and maneuver classifier in more detail. The maneuver parameter generator is implemented by the derived states computer, to generate from the estimated states an estimate of the threat tactics. This is accomplished by postulating the line-of-sight (LOS) intercept guidance model introduced earlier by (4.2b), and estimating two parameters: the guidance gain K, and a parameter α reflecting the alignment of the threat acceleration $(\hat{\mathbf{a}})$ with the expected acceleration $(\mathbf{a}_c)$ under LOS guidance. Thus, if an unknown has a state history that yields a (K,α) a pair with K in the range expected for threatening LOS guidance, and with α near unity, and thus consistent with an LOS guidance heading, the classifier would conclude that the unknown was a threat. Conversely, an inconsistent (K,α) pair (such as occurs under zero acceleration) would lead to a conclusion that the unknown was a friendly.

Actual classification into threat/friendly classes on the basis of the maneuver *features* (K,α) is accomplished by the neural net classifier illustrated in figure 4.8. Here we show the topology of the network used in modeling the threat/friendly classification function. The network consists of an input layer, one hidden layer, and one output layer. The input layer consists of 11 node pairs, with each pair dedicated to a specific (K, α) guidance parameter pair, estimated at a

fixed point in time. As the engagement proceeds, the parameter pairs *slide down the* input nodes from top to bottom, so that the network effectively *windows* the input data (over an 11-point window) and dynamically updates its outputs over the course of the engagement. The two output nodes shown are the classifier outputs: a unity value at a given node indicates the network's classification decision, and a zero indicates the network's rejection of that classification. In effect, the threat node defines the desired threat occurrence probability $P\{e_i\}$ of (4.18), while the friendly now defines its complement. Although only two output nodes are shown, it is a direct matter to logically combine the outputs to determine if neither node output is sufficiently *compelling* to declare a classification, so that the classifier should declare an *unknown*, and continue to monitor the trajectory parameters. Note that the sliding window input to the network classifier results in an *ongoing* dynamic classification of each observed object in the tactical space, a feature which goes significantly beyond a simple *one-shot* static classification of potential threats. The network was trained on four pairs of friendly/threat maneuver patterns. Training lasted for 700 cycles an achieved an RMS error level of 0.07.

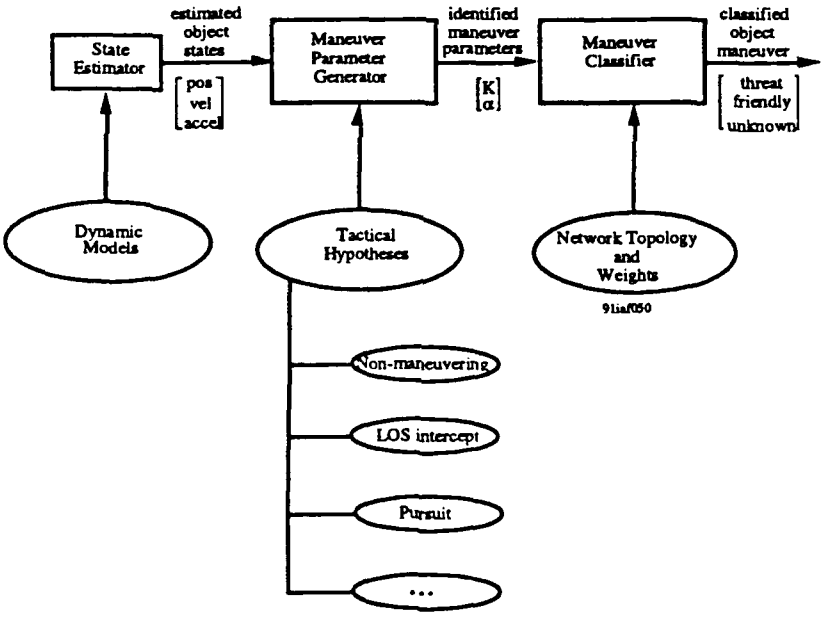
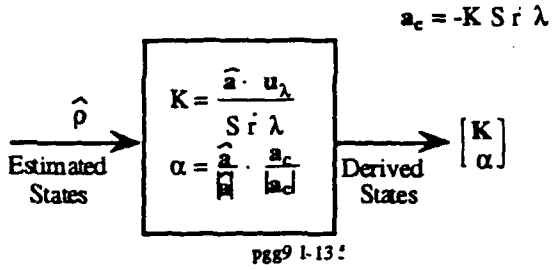


Figure 4.7 Threat Maneuver Classification for Tactical Situation Assessment

- **Derived States Estimator**



- **ANN SA Classifier**

  - Occurrence Probabilities

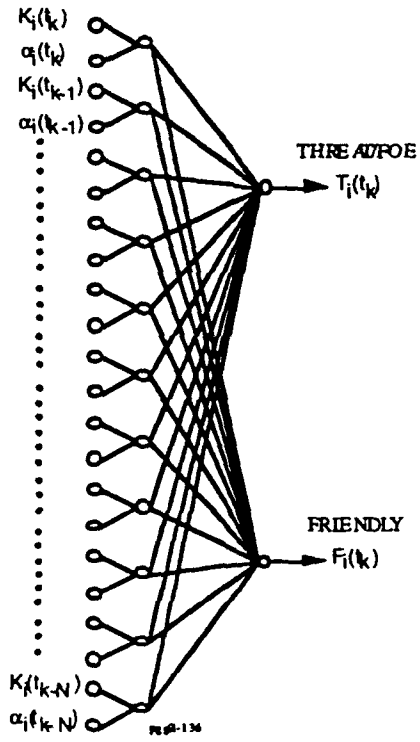


Figure 4.8: Maneuver Parameter Generator and Maneuver Classifier

Although all other components of the situation vector could be classified in a similar fashion, we chose a simpler representation, based on a direct assessment of probabilities, based on the estimated object states generated by the state estimator.

In particular, to generate the probability associated with a threat object being high priority (recall (4.17)), we used the following assessment function:

$$P\{f_i\} = P\{\rho_i = \min (\text{all } \rho_j | \hat{\rho}_j, \sigma_{\rho_j}\} \qquad (j=1...N) \tag{4.19}$$

Here, we calculate the probability that the ith object is the closest of all N objects on the TSD screen, based on the *estimated* range $\hat{\rho}_j$ and the associated variances σ_{ρ_j} maintained by the pilot estimator model. Naturally, the accuracy of this priority assessment will depend on the accuracy of the estimated (range, variance) pairs $(\hat{\rho}_j, \sigma_{\rho_j})$.

Finally, we model the assessment of being within the preferred FPS envelope (recall (4.17)) as a simple range dependent assessment, via:

$$P\{g_i\} = P\{\rho_i < \rho_{FPS} | \rho_i, \sigma_{\rho_i}\} \tag{4.20}$$

Here, we calculate the probability that the ith object is within the FPS range for firing a missile, or ρ_{FPS} . Expansion to a full azimuth-dependent envelope involves coding ρ_{FPS} as $\rho_{FPS}(\psi)$, but this was not done for this initial feasibility modeling effort. In effect, we assume a circular FPS envelope.

Once the three occurrence probabilities are computed, the assessed situation vector $\hat{S}_i$ can be specified via (4.18b) introduced earlier. The *disparities* between this assessed situation and the actual situation can then be defined functionally via the following normed difference:

$$SD_i(t) = \left| S_i(t) - \hat{S}_i(t)/M^{1/2} \right| \tag{4.21a}$$

where M, the dimension of S, is 3, and is introduced to ensure that SD_i stays in the [0, 1] range. This then defines the SD for a *single* object at a given time t in the engagement. We can then define the disparity across all objects (i = 1, ...N) at time t via:

$$SD(t) = \sqrt{\frac{1}{N} \sum_i SD_i^2(t)} \tag{4.21b}$$

Finally, we can define the SD across the entire engagement via a simple integration over the duration of the engagement time T, via:

$$SD = \frac{1}{T} \int SD(t)\, dt \tag{4.21c}$$

Thus, we can define the situational disparity SD across the full engagement, and thus, via its inverse, SA, across the engagement.

4.3.4 Decision Maker

The decision-maker module selects an appropriate action as dictated by the assessed situation. For our implementation we have used a simple rule-based decision maker. Figure 4.9 shows the three rules implemented in the decision maker. The first rule declares an object a threat, if its occurrence probability of being a threat $(P\{e_i\})$ is greater than some threshold e^{max}. The second rule determines prioritization of threats. If the object is a threat and its prioritization occurrence probability $(P\{f_i\})$ exceeds all other prioritization occurrence probabilities, then the object is asserted a high priority threat. Finally, the third rule declares that if the object is a threat and its probability that it is within the FPS $(P\{g_i\})$ is greater than a pre-set threshold, then a decision to fire a missile is made. These simple production rules thus define the actions following object classification, threat prioritization, and FPS declaration.

- **Rule-Based Decision Making**

  - Object Classification

    IF

    THEN object (i) = "threat"

  - Prioritization

    IF object (i) = "threat"

    and

    THEN object (i) = "high priority threat"

  - FPS

    IF object (i) = "threat"

    and

    THEN "fire missile"

Figure 4.9: Decision Maker Rule-Base

4.3.5 Procedure Effector

The procedure effector module selects the appropriate action based on the decision reached by the decision-maker module. For our implementation, the available procedures to choose from are: a) guidance law selection; b) control actions; and c) weapons mount. The guidance selection involves choosing a LOS intercept for identified high priority threats. If no high priority threats exist, a cruise mode (i.e., constant velocity) is selected. Figure 4.10 shows a block diagram of the control procedure implemented for a LOS intercept. The procedure transforms the desired accelerator command ($\mathbf{a}_c$) into the vehicle control triplet of (n_x n_z ϕ). Finally, the procedure associated with the weapons mount is to fire a missile once a decision that a threat is within FPS has been asserted by the decision-maker module.

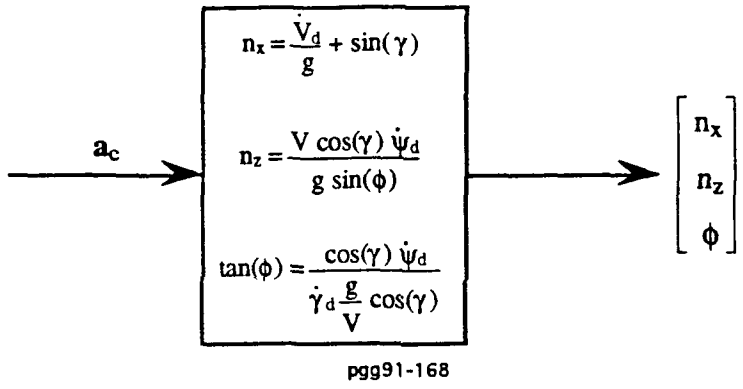


Figure 4.10 Implementation of Control Procedure Effector

5. SITUATION DISPLAY EVALUATION

This chapter presents the results of our proof-of-concept demonstration of the model-based method. It focuses on the analysis of a tactical situation display. Section 5.1 presents results for a baseline scenario and a nominal baseline display. Section 5.2 presents results for variations in the baseline format and for several different display enhancement options.

5.3 Baseline Scenario

Consider the tactical situation shown in the 3 x 5 mile gaming area sketched in figure 5.1a. Here, ownship is shown in the center, flying straight and level due east at 104 kts. At the beginning of the engagement, three unknown objects are in the gaming area: object 1 due north of ownship and heading due south at a high speed of 154 kts; object 2 southwest of ownship and headed northeast at 111 kts, also towards the ownship; and object 3 behind ownship headed due north at 88 kts. During the course of the engagement shown in figure 5.1b, objects 1 and 2 engage in pursuit trajectories designed for an interception and short-range missile firing at ownship; object 3 maintains a non-maneuvering non-threatening course throughout the engagement.

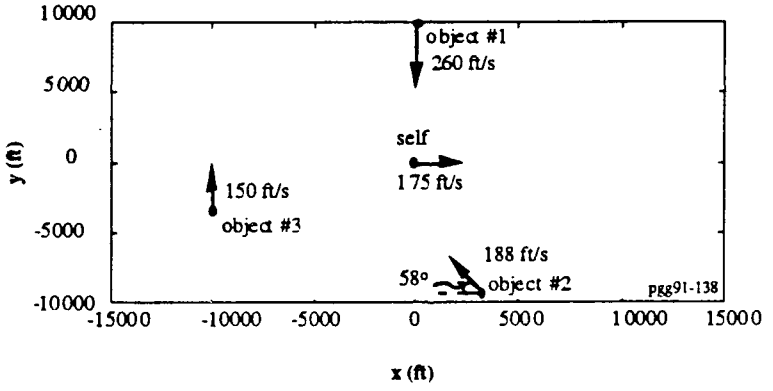


Figure 5.1a: God's-Eye-View of Baseline Scenario: Initial Engagement Conditions

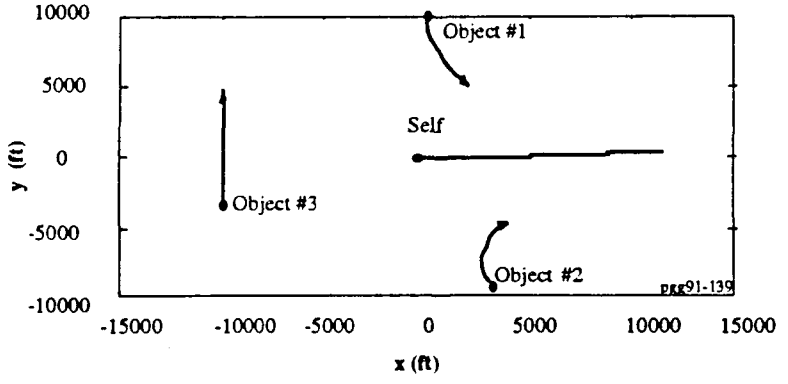


Figure 5.1b: God's-Eye-View of Baseline Scenario: Complete Engagement Trajectory

From the beginning of the simulation, one crewmember monitors a radar display that provides relative position and velocity estimates of all three objects; the display, in turn, is driven by the hybrid sensor model sketched earlier in figure 4.2, incorporating realistic sensor noise levels to simulate operational limitations in the radar and the display. Specifically, we set sensor noise levels on range and range rate to be 5%, and azimuth and azimuth rate noise levels were set to 5 deg and .5 deg/sec, respectively. Given this noisy *relative-state* information, the first objective of the crewmember is to estimate the god's-eye-view of the situation, and, in particular, estimate the trajectory *signatures* that reveal which object is a threat, and which is a friendly.

Figure 5.2 shows the ***internalized*** god's-eye-view estimates maintained by the crewmember for each of the three objects, plotted separately on expanded scales to enhance the difference between the actual paths (solid lines) and the estimated paths (dashed lines). Note that object 1 position is estimated fairly accurately throughout the engagement; object 2 is poorly tracked initially, but estimation errors quickly grow smaller with time; object 3 is also fairly well-tracked throughout the engagement. Thus, the crewmember has a fairly accurate mental picture of the locations, over time, of all three objects. He also has a fairly accurate estimate of relative range to all three objects as illustrated in figure 5.3.

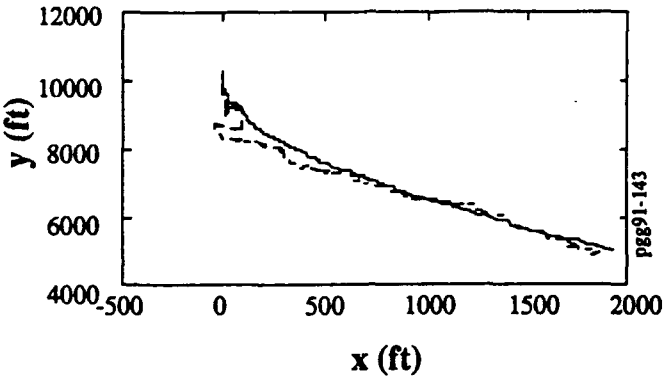


Figure 5.2a: State Estimates — Object 1

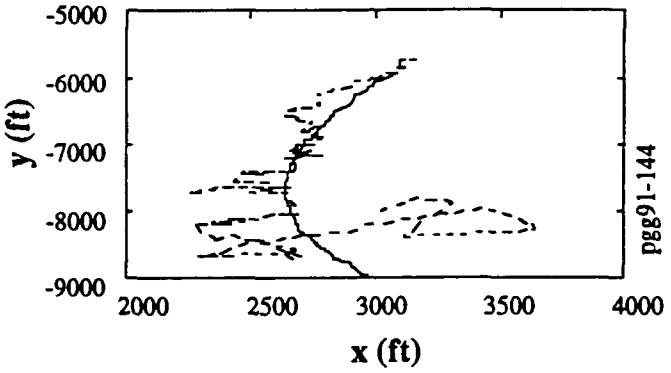


Figure 5.2b: State Estimates — Object 2

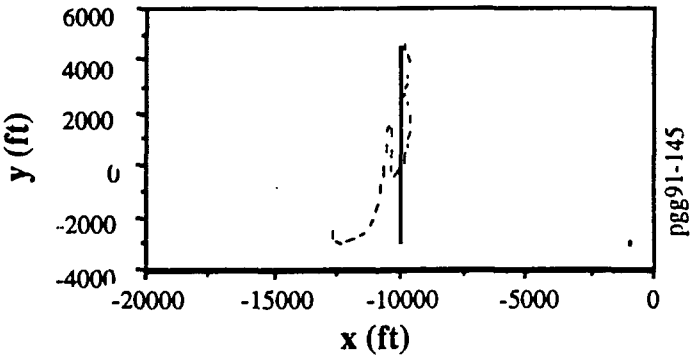


Figure 5.2c: State Estimates — Object 3

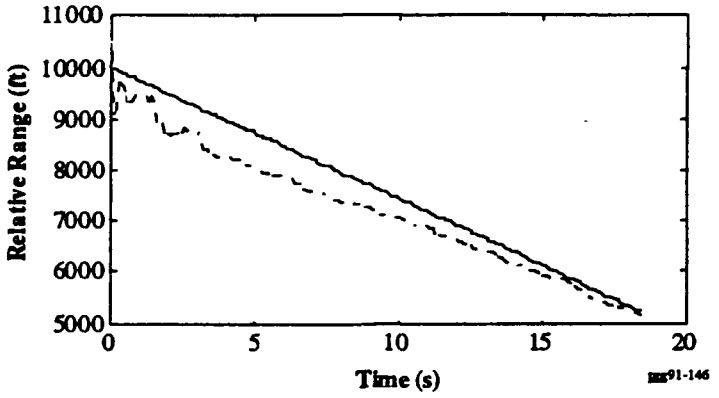


Figure 5.3a: Relative Range for Object 1

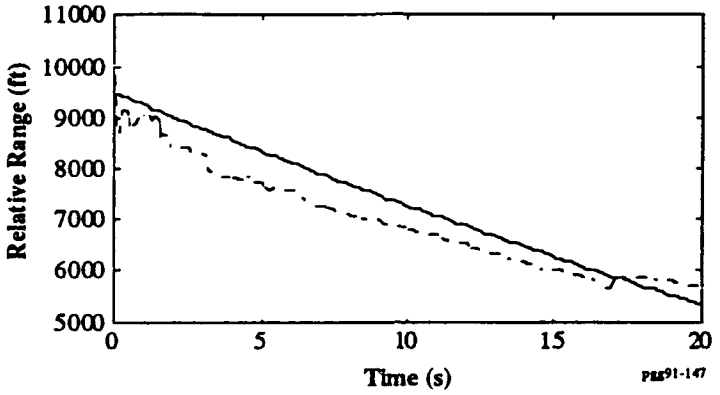


Figure 5.3b: Relative Range for Object 2

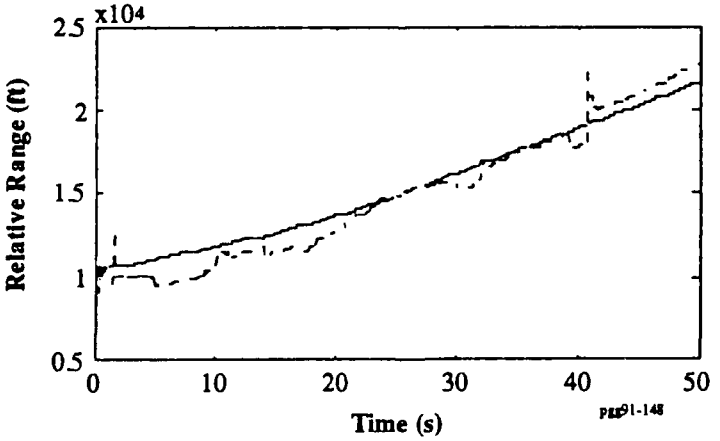


Figure 5.3c: Relative Range for Object 3

In generating the position estimates of figure 5.2, the crewmember also generates *velocity* estimates, (not shown), which, when combined appropriately, are used to infer the offensive maneuver parameters of each of the three objects. This is accomplished by hypothesizing an intercept guidance logic and, in effect, estimating how closely the perceived object behavior

conforms with behavior that would be expected from an attacking (closing) threat. In terms of the guidance logic described in section 4.2.1, this amounts to an expectation of:

- an estimated positive guidance gain K which stays in the 1 to 10 range
- an error in computed heading α which stays relatively small, on the order of 10 deg or less.

To accomplish this assessment requires the computation of the derived guidance parameters (K, α) for each object in the scenario. These are illustrated in figure 5.4. Note how objects 1 and 2 (the threats) show large K values, whereas object 3 (friendly) does not.

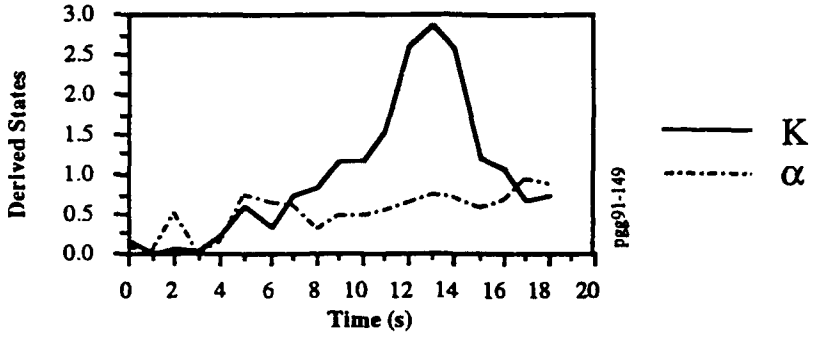


Figure 5.4a: Derived Guidance Parameters for Object 1

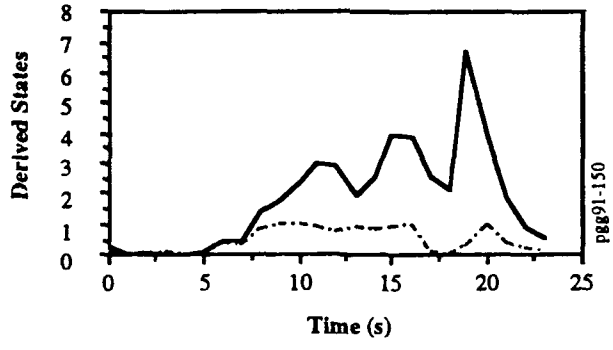


Figure 5.4b: Derived Guidance Parameters for Object 2

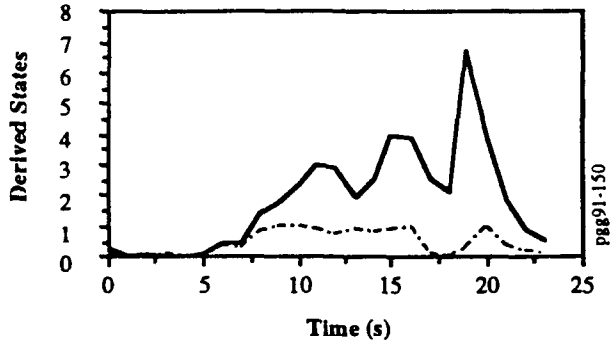


Figure 5.4c: Derived Guidance Parameters for Object 3

Given these guidance parameter histories, the SA function is then formally modeled by the network's classification of these histories, in effect, the network's recognition of *threatening object* acceleration signatures. Here, the net result is shown in figure 5.5, which plots the crewmember's internally-generated confidence estimate given by the probability occurence $P\{e_i\}$ that the ith object is indeed a threat. Here a zero value indicates that it is a friendly, while a value of unity indicates that it is a threat. Note that all three confidence estimates start at zero because none of the trajectories evidence the signature acceleration behavior of a threat early on. Later however, objects 1 and 2 start behaving like threats, and the corresponding confidence values ($P\{e_1\}$and $P\{e_2\}$) rise towards unity accordingly. Object 3 never does behave in a threatening fashion, so that its acceleration profile fails to disclose any such threatening signature and the corresponding confidence level remains low, as seen in figure 5.5.

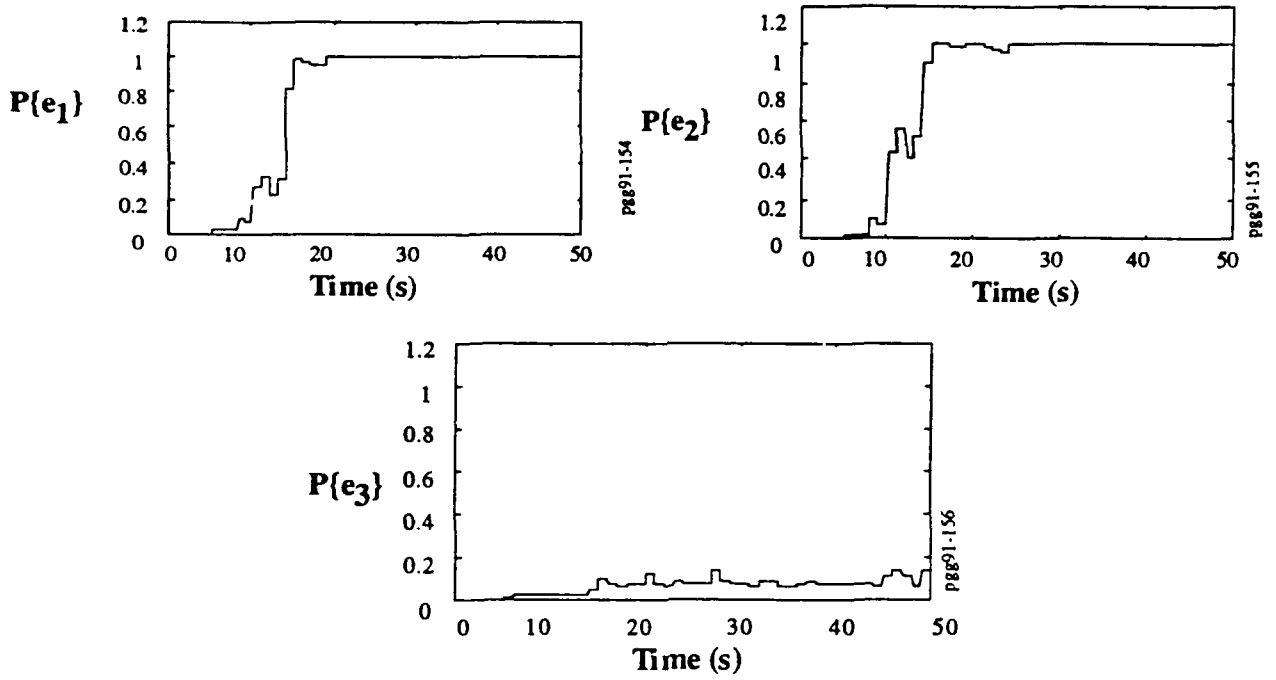


Figure 5.5: Threat Assessment Over Time

Once the confidence level for a given object exceeds a pre-set minimum level (in our case, we chose a level of 0.60), the crewmember declares an object a threat. In the scenario described here, reference to figure 5.5 shows that the crewmember declares objects 1 and 2 to be threats at about 15 sec into the engagement; object 3 is never declared a threat by the crewmember.

Similar confidence estimates for other situational attributes are maintained by the crewmember. Two others that are critical deal with threat range and location of the threat within the preferred missile firing point. Threat range is needed to determine which threat is the closest,

so that the ownship can go at the most threatening adversary. This assessment is based on the range estimates computed by the upstream information processor, and illustrated in figure 5.3 above. The probability calculation is as described earlier in (4.19). Figure 5.6 shows the resulting prioritization assessments, for each of the three objects, with the dashed line indicating the assessment, and the solid line the actual priority (based on closest range). Note the correct and rapid initial assessment for object 2, and then the switch at 11 sec to object 1, again a correct assessment. The switch back to object 2 follows after firing at and eliminating object 1 (see below). Finally, note the eventual switch to object 3, the last among the high priority objects.

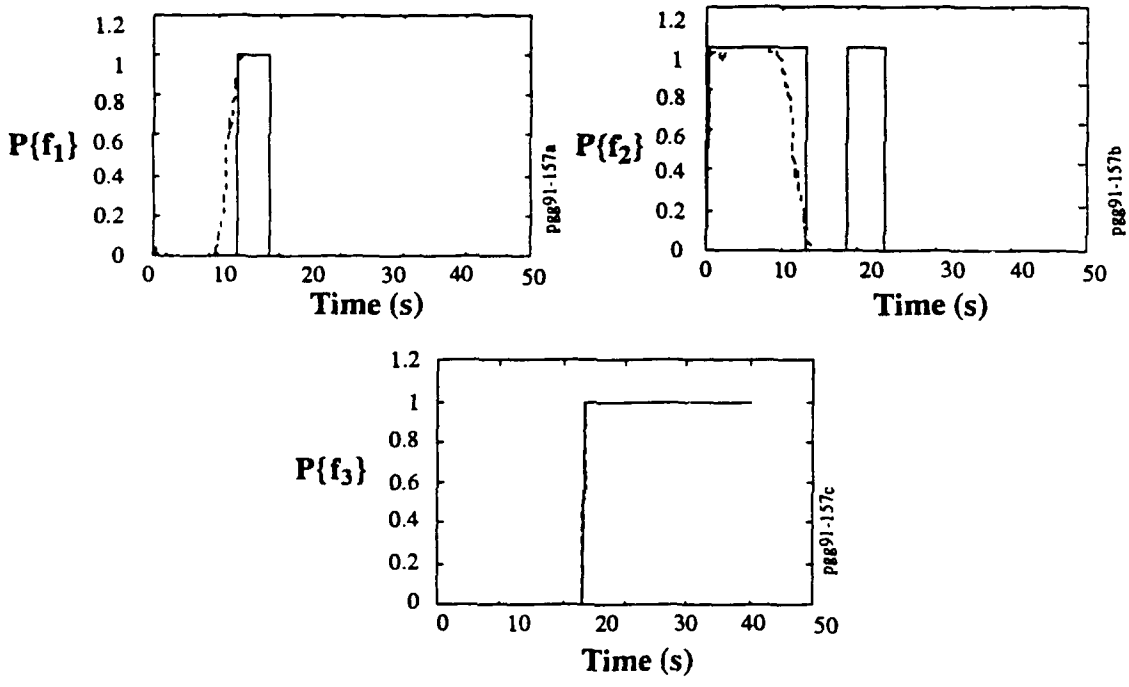


Figure 5.6: Priority Assessment Over Time

Threat location in the weapons envelope is computed on the basis of the threat's estimated relative position vector and the known kill envelope of the ownship missile. This is computed as described earlier by (4.20). Figure 5.7 shows the resulting fire point selection (FPS) assessment, for each of the three objects. Again, the dashed line indicates the assessment, and the solid line the actual FPS situation (based on location within the weapons envelope). Note the early assessment at 17 sec (and 5,000 ft range) for object 1, which is then fired upon and eliminated. This leads to the rapid reallocation of attention to object 2, and subsequent firing (at 23 sec) upon it as well. Finally, note that FPS assessment is always null for object 3, since it is not perceived as a threat, on the basis of the pilot's threat designation.

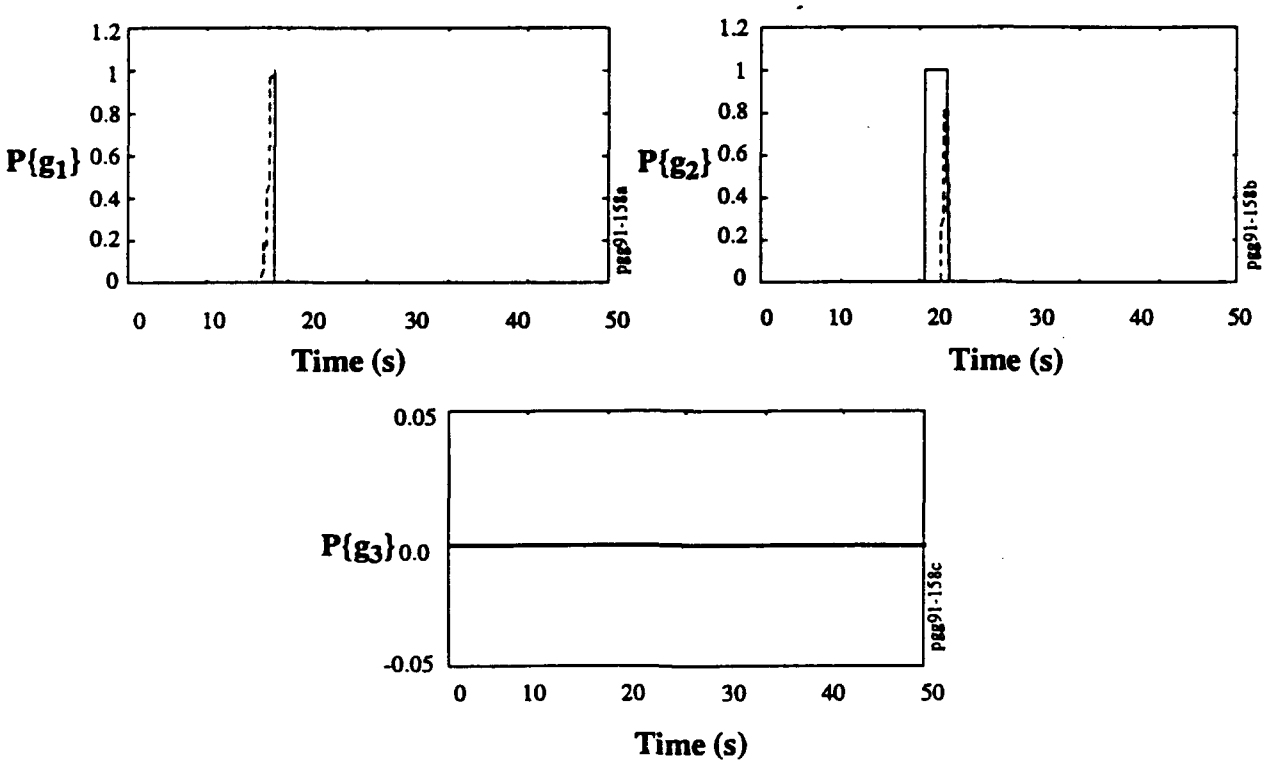


Figure 5.7: Fire Point Selection (FPS) Assessment Over Time

If we consider the three situational components of figures 5.5, 5.6, and 5.7 as components of the three-dimensional assessed situation vector $\hat{\mathbf{S}}$ (as defined earlier by (4.17b)), we then have a single SA vector defining object type, relative closeness, and kill potential P_K. By differencing this internally estimated situation with the actual situation **S**, we obtain a metric of the crewmember's *situational disparity* (SD) over time (defined formally by (4.21b)), as it evolves over the engagement. A low value indicates low disparity and an accurate assessment of the situation by the crewmember; a high value conversely indicates an incorrect assessment. The result is shown in figure 5.8 which shows how the situational disparity is high early on because both threats are incorrectly assessed as friendlies, but then rapidly drops in the 10 to 20 second time frame as the crewmember processes the display data, updates his assessments regarding object type, range, and P_K potential, and converges on an accurate assessment of the situation. Three peaks are apparent in the SD history. The first is at 12 sec and is due to the fact that object 1 is incorrectly assessed as being higher priority than object 2, when in fact the reverse is true for a few seconds of the engagement. The second peak occurs at 18 sec when object 1 is incorrectly assessed as being with the FPS range, when it is just outside of the FPS range, for the few seconds of the peak shown in the plot. The final peak occurs at 21 sec, which is when object 2 comes within FPS range, and is not eliminated for several seconds.

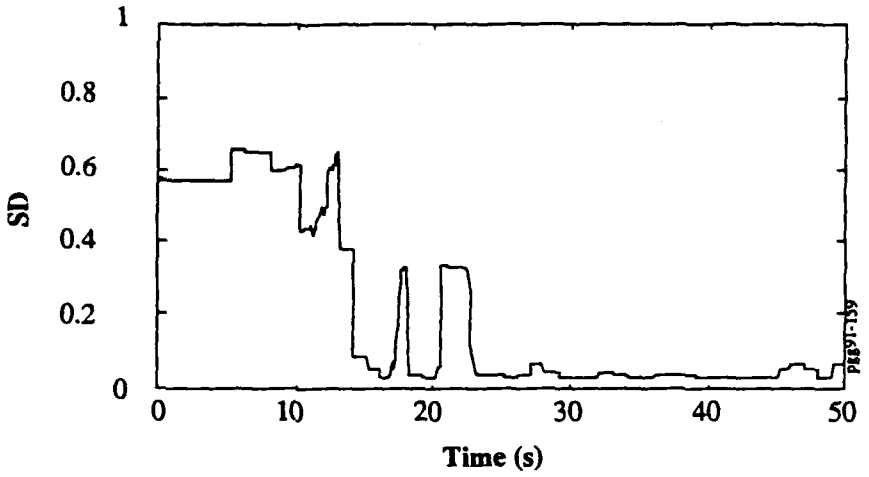


Figure 5.8: Situational Disparity Time History (Composite Metric)

Once this situational disparity time history is available, it is a direct matter to compute a scalar situation awareness metric characterizing the overall engagement. The metric $\overline{S}$ is simply computed as the root-mean-square (RMS) value, over time, of the situational disparity history. As noted earlier, this then allows us to compare one engagement vs another, in terms of this internal awareness metric. We will illustrate this comparison in the next section.

5.2 Analysis of Display Enhancement Options

We now demonstrate the use of the situation awareness metric for evaluating enhancement options for the nominal situation display, including display format and display decision-aiding. The display format factors cover symbol size and map scale selection and are described in section 5.2.1. The display decision aids cover identification friend or foe (IFF), target prioritization, fire point selection (FPS), and attention focusing; these are discussed in section 5.2.2

5.2.1 Display Format Analysis

We have already detailed the nominal display format in section 4.22. On the basis of the analysis described there, the nominal symbol size and map scale result in a display noise level specified by range and range rate noises at the 5% level and azimuth and azimuth rate noises at the 5° and 0.5°/sec level. Table 5.1 illustrates the effect of varying the nominal symbol size (S) and the nominal operating range (R). The nominal (R, S) pair yields a 5% and 5° nominal noise level, as shown in the center of the table. Increasing or decreasing the operating range R yields a corresponding increase or decrease in the associated threshold as shown by the center row of the table. Likewise, increasing or decreasing the symbol size S shows a proportionate increase or decrease of the threshold level along the center column of the table.

Table 5.1: Impact of Symbol Size and Map Scale (Operating Range) on Display Noise Level

Operating Range / Symbol Size	.25 R	.5 R	R	2 R	4 R
.25 S	0.31	.63	1.25	2.50	5.00
.5 S	0.63	1.25	2.50	5.00	10.00
S	1.25	2.50	5.00	10.00	20.00
2 S	2.50	5.00	10.00	20.00	40.0
4 S	5.00	10.00	20.00	40.0	80.00

p89-160

Units of % for range and ° for ψ

To evaluate the impact of symbol size and map scale on overall situation disparity (SD) during the engagement, several simulation runs were conducted in the fashion described in the previous section. The results are illustrated in figure 5.9, which shows situation disparity plotted as a function of display noise level. Note that the nominal display noise level yields an SD level of approximately 0.20. An increase in the display noise level, corresponding to either an increase in the map scale or an increase in symbol size, yields a corresponding increase in the SD level. In particular, a doubling of display noise level yields a 75% increase in SD; a quadrupling of display noise level yields a 100% increase. In contrast, a reduction of the noise level to 1/2 or 1/4 of the nominal results in effectively no difference from the SD obtained nominally. Clearly, the nominal symbol size and map scale puts us at the knee of the curve in terms of SD levels.

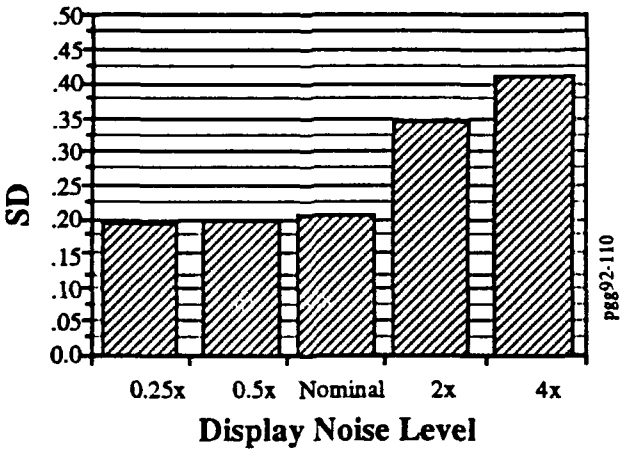


Figure 5.9: Engagement Situational Disparity as a Function of Display Noise Level

5.2.2 *Decision Aid Analysis*

We now describe model-based analysis of four different decision aids: IFF tagging, target prioritization, fire point selection, and attention focusing.

Figure 5.10 illustrates the impact of an IFF display aid on overall SD. Here it is assumed that the system provides, in accordance with figure 4.4, symbology to indicate the current status of each object in the display area of interest: friendly, hostile, or unknown. This functional task reallocation between man and machine is modeled by assuming: 1) perfect performance on the part of the IFF tagging system; 2) weighting of the IFF indication with that generated internally by the crewmember; and 3) a variable confidence level in the aid held by the crewmember, which determines the weighting, in accordance with:

$$P^*\{e_i\}=(1-\sigma_e)P\{e_i\}+\sigma_e P_{IFF}\{e_i\} \quad (5.1)$$

where P^* is the resulting threat assessment probability, P is that generated by the crewmember *without* the aid, and P_{IFF} is that generated by the IFF aid. Confidence weighting is determined by σ_e, with a 100% value indicating full confidence in the aid.

The figure shows *nominal* unaided performance (where confidence = 0%) yielding the lowest SA (i.e., the highest SD). It also shows how SA improves (the SD metric drops) as the crewmembers' confidence in the IFF aid grows from 0% to 100%. A linear growth trend in SA is seen with increasing confidence, although it should be clear that SD will not drop to 0 since IFF performance is only one component of overall SA. It is important to note that the reallocation of the IFF function from crewmember to on-board aid can improve SA, but only if the crewmember has sufficient confidence in the reliability of the IFF aid.

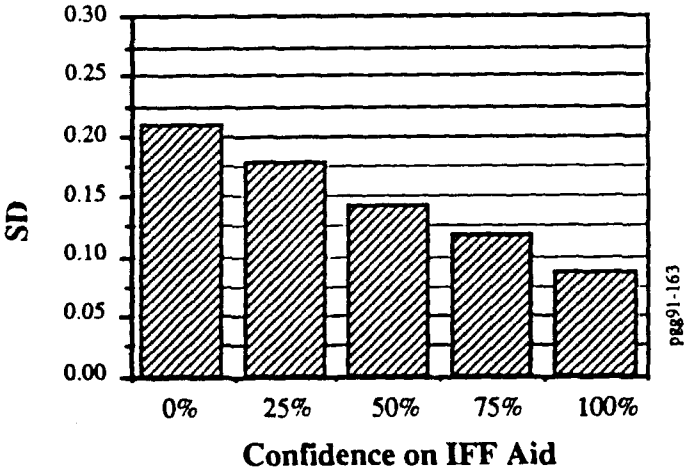


Figure 5.10: SD Trends with IFF Aiding

Figure 5.11a illustrates the effect of priority aiding on overall crew SD. Here the target prioritization aid is assumed to provide the crewmember with an indication of object priority via color coding of each tagged object indicated on the display. Again, three assumptions are made: 1) perfect performance on the part of the priority aid; 2) weighting of the priority aid indication with that generated internally by the crewmember; and 3) a variable confidence level in the aid held by the crewmember, which determines the weighting, in accordance with:

$$P^*\{f_i\}=(1-\sigma_f)P\{f_i\}+\sigma_f P_{PRIO}\{f_i\} \quad (5.2)$$

where P* is the resulting priority assessment held by the crewmember, P is that held by him *without* the aid, and P_{PRIO} is that generated by the aid. Again, confidence weighting is determined by σ_f.

The figure shows the *nominal* unaided performance (again where the confidence = 0%), and the effect of increasing confidence levels, to a 100% level. Clearly, the prioritization aid has little effect on overall SD. The implication here is that target priority is obvious when only three objects are present and in the scenario.

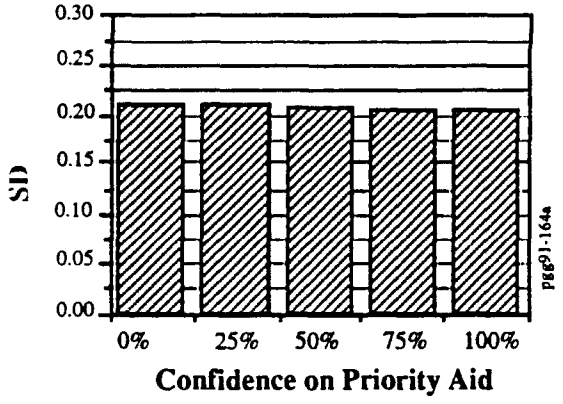


Figure 5.11a: SD Trends with Priority Aiding (Three Object Scenario)

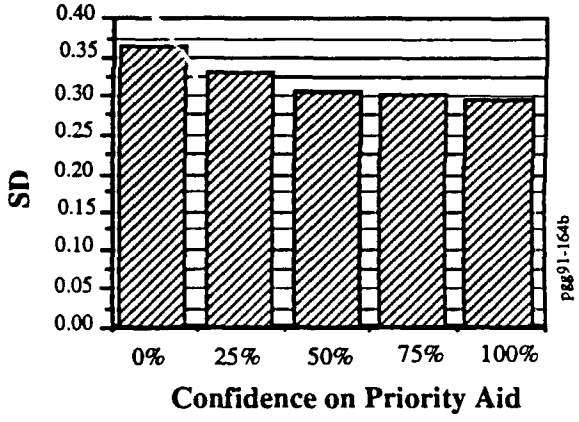


Figure 5.11b: SD Trends with Priority Aiding (Six Object Scenario)

To evaluate the impact of priority aiding in a more complex scenario, we simulated a six-object scenario as illustrated in figure 5.12. Here, objects 1, 2, 4, and 5 are threats, whereas objects 3 and 6 are friendlies. Following the simulation, the SD metrics were computed as before, this time across all six objects. The results are illustrated in Figure 5.11b. In this scenario we see an overall decrease in SA, because of the greater demands of the six-object task, but we also see a clear improvement with increased confidence in priority aiding, with an approximately 15% improvement going from the 0% confidence level to the 100% confidence level. However, there is a clear marginal return with increasing confidence level. Finally, it is appropriate to note that

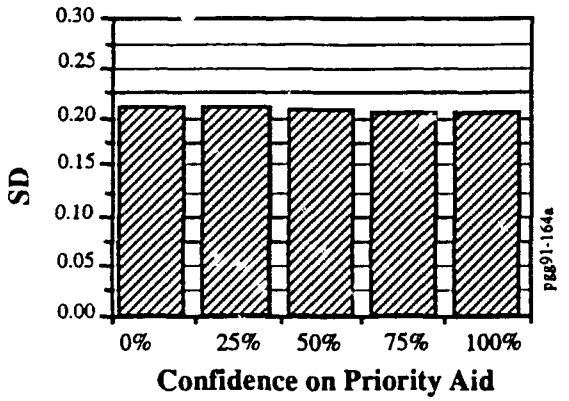


Figure 5.11a: SD Trends with Priority Aiding (Three Object Scenario)

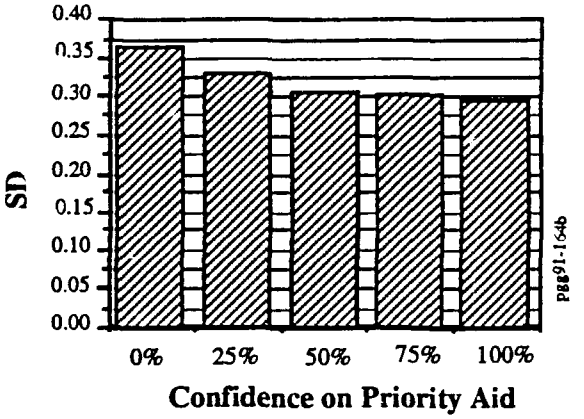


Figure 5.11b: SD Trends with Priority Aiding (Six Object Scenario)

To evaluate the impact of priority aiding in a more complex scenario, we simulated a six-object scenario as illustrated in figure 5.12. Here, objects 1, 2, 4, and 5 are threats, whereas objects 3 and 6 are friendlies. Following the simulation, the SD metrics were computed as before, this time across all six objects. The results are illustrated in Figure 5.11b. In this scenario we see an overall decrease in SA, because of the greater demands of the six-object task, but we also see a clear improvement with increased confidence in priority aiding, with an approximately 15% improvement going from the 0% confidence level to the 100% confidence level. However, there is a clear marginal return with increasing confidence level. Finally, it is appropriate to note that the potential improvement in SA can be expected to be greater with additional objects beyond the three and six contemplated here. We would also expect the display aid payoff to improve with additional clutter on the display, since the decision aid would serve to reduce attention-related noise on the part of the operator.

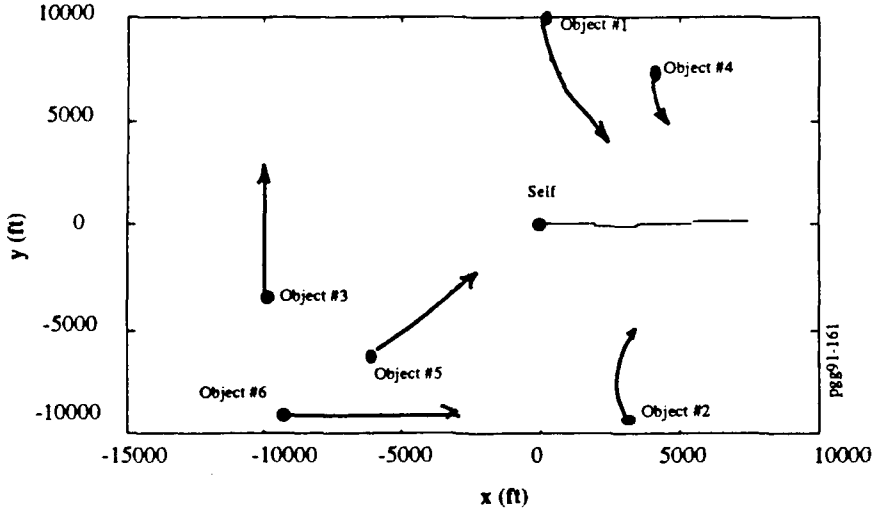


Figure 5.12: Six-Object Scenario

Figure 5.13 illustrates the effect of FPS aiding on SD trends. As with the other two aids, weighting of the aid is in accordance with:

$$P^*\{g_i\}=(1-\sigma_g)P\{g_i\}+\sigma_g P_{FPS}\{g_i\} \quad (5.3)$$

where P* is the resulting FPS assessment held by the crewmember, P is that held by him without the aid, and P_{FPS} is that generated by the aid. Confidence weighting is determined by σ_g. Figure 5.13a shows the impact of FPS aiding for the three-object scenario. As with the priority aiding just discussed, the FPS aiding has little impact on overall SD as confidence in that aid increases. Again, this can be attributed to the fact that the three-object scenario can be dealt with directly by the pilot without aiding. However, when the six-object scenario is evaluated, the results are shown as in figure 5.13b. Here we again seee a decrease in SA overall, but we also see a clear improvement in SA as the FPS aid confidence level increases, with an overall 40% improvement in SA when full confidence is placed on the FPS aid. The improvement appears to be relatively linear with increasing confidence. Again, there exists a potential for even greater effectiveness of the FPS aid when the crewmember is either faced with more objects than the six-object scenario considered here, or when there exists more clutter on the display.

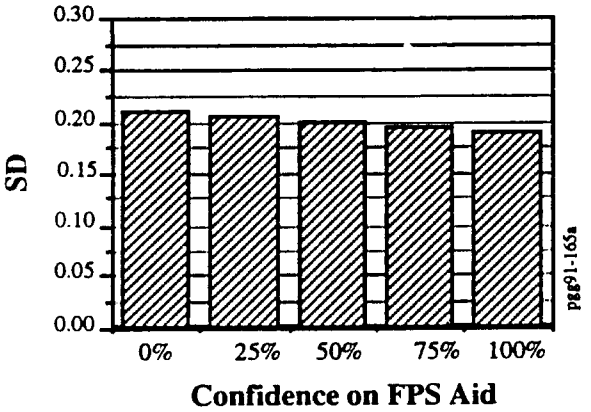


Figure 5.13a: SD Trends with FPS Aiding (Three-Object Scenario)

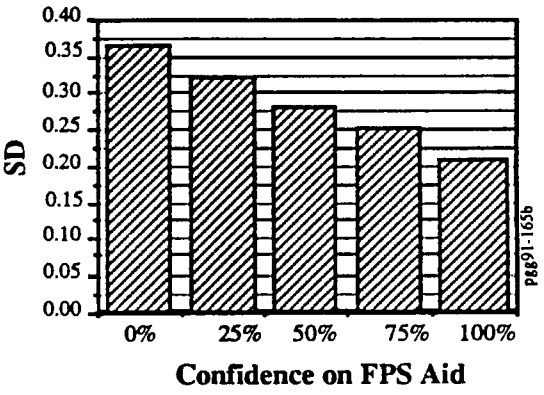


Figure 5.13b: SD Trends with FPS Aiding (Six-Object Scenario)

The final display aid to be evaluated was a focusing aid which selectively blanked out low priority objects and forced the crewmember to focus on the high priority objects presented on the display. Figure 5.14 compares crew SA under the baseline unaided case with SA associated with use of the focusing aid. Here we see a slight *decrease* in SA (an increase in SD) when the focusing aid is used, clearly an unacceptable situation for a display enhancement. This unexpected effect can be attributed to the loss of information on low-priority objects, which is greater than the improved SA due to the attentional focus on high priority targets. Thus an

overall net loss in SA results. The impact of the focusing aid is thus negative in this situation, although additional simulation runs would be required to determine whether this conclusion holds up under the majority of scenarios.

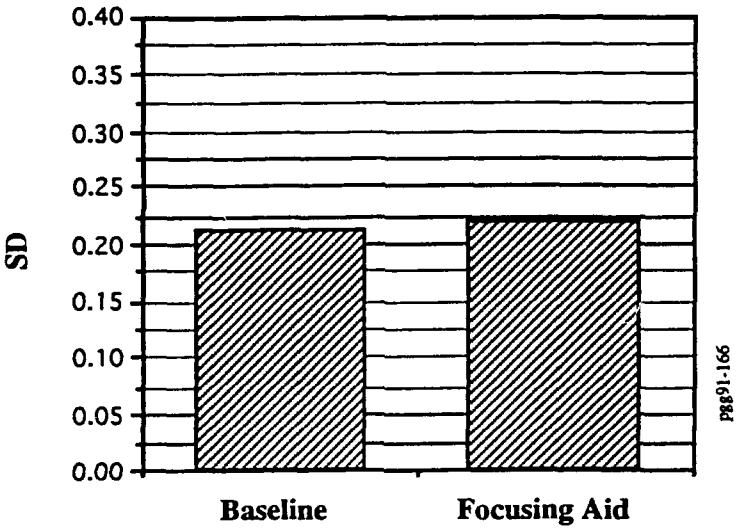


Figure 5.14: SD Trends with Attention Focusing Aid (Three-Object Scenario)

5.2.3 *Summary of Display Enhancement Analysis*

Table 5.2 summarizes the results of the model-based analysis. The simulation evaluating the effect of display resolution and map scale range shows that the nominal choice of symbol size and scale range places the crewmember at the knee of the curve defining SA as a function of display noise. A decrease in SA is seen with lower resolution display components: a 75% increase in SD results when the noise is doubled, and a 100% increase results when the noise is quadrupled. No change in SA occurs with higher resolution, that is, when display noise is reduced.

The studies with the IFF display aid show a 50% increase in SA, as crewmember confidence on the IFF display increases to 100%. A general linear progression in SA occurs with increasing confidence on the aid.

Table 5.2: Summary of Model-Based Analysis

- **Display Resolution / Map Scale Range**
  - Nominal at "knee" of curve
  - Increase in SD with lower resolution
    - 75% increase for 2x
    - 100% increase for 4x
  - No change with higher resolution
- **IFF Display Aid**
  - 50% decrease in SD as confidence goes to 100%
- **Priority Display Aid**
  - Little effect for 3 object scenario
  - 15% improvement for 6 object scenario
  - potential for more improvement with added clutter/objects
- **FPS Display Aid**
  - Little effect for 3 object scenario
  - 40% improvement for 6 object scenario
  - potential for more improvement with added clutter / objects
- **Focusing Aid**
  - Little effect due to display noise

The study with the priority display aid shows that there is little effect under the nominal three-object scenario, with SA remaining approximately constant across all levels of priority aid confidence. However, when the priority aid is evaluated in a higher tempo six-object scenario, a 15% improvement in SA is seen when confidence in the priority aid goes to 100%. However, a marginal return in improving SA is seen with increasing confidence on the priority aid. It can be expected that the priority aid will further improve SA under situations in which there are objects added to the scenario and/or clutter serves to make the display busier.

The study of the FPS display aid also displayed little effect for the three-object scenario with SA remaining relatively constant across all levels of confidence on the FPS aid. However, a 40% improvement is seen in SA for a higher tempo six-object scenario, with an approximately linear improvement in SA as the confidence level increases. Again, as with the priority display aid, we anticipate that there exists a greater potential for SA improvement under scenario conditions in which more objects are in the gaming area and/or clutter is added to the display.

Finally, the study of the focusing aid showed little effect when the focusing is used to enhance the display. In fact, a slight negative effect was found. This is due to the fact that focusing reduces information of low priority targets, information which is clearly contributing to the crewmember's overall SA of the entire scenario.

These results and others generated under the Phase I effort demonstrate that reasonably realistic event timelines can be generated with a fairly limited procedure rule base, and that the dynamic evolution of the pilot's situational assessment, decision-making, and task performance can be closely followed throughout a mission phase and engagement. The results also show how simple metrics can be generated for internal model states and external world states, to support evaluations of situational disparity and the appropriateness of overt decision choices.

These simulator capabilities also demonstrate how a number of engagement and crew variables affect engagement outcome. We can directly study the effects of: 1) initial engagement geometry; 2) sensor inaccuracies; 3) pilot-related limitations in sensory/perceptual processing; 4) internal state estimation strategy; 5) decision criteria in situation assessment assertions and 6) decision-making procedure definitions. A wide variety of factors can be evaluated over a range of tactically relevant values.

6. SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

6.1 Summary

Our general technical approach to evaluating feasibility of the SA display analysis design method centered on: 1) specification of an integrated functional model of the crew/system that represents the system relations, the pilot activities, and provides "hooks" to the crew's internal states, assessed situations, and decisions; 2) assessment of enabling technologies for crew/system model development, including modern estimation (ME) techniques, artificial neural networks (ANNs), and expert systems (ES) technology; 3) implementation of a limited scope version of the model for a display design evaluation and assessment of afforded SA; 4) conduct of a proof-of-concept demonstration to evaluate SA as a function of display format and aids; and 5) identification of design tool objectives and development paths for full-scope implementation and validation.

We specified an integrated functional crew/system model (CSIM) that allows us to combine and integrate system-related and human-related components that drive overall performance and crew SA during an engagement. This model provides an architecture for integrating the crewmembers' basic functions of: 1) sensory/perceptual processing of the display interface cues; 2) information processing (IP) of continuous vehicle states and discrete cueing aids; 3) situation assessment (SA) for driving engagement-relevant decisions; 4) decision-making (DM) for selecting among alternative actions; and 5) procedure execution for effecting guidance and missile firing commands.

We described a narrow range of crew procedures that might be evaluated within this framework, and identified candidate model-based SA metrics for display evaluation studies.

In the specification of the CSIM model, we evaluated enabling technologies for full-scale development. For the sensory processing, information fusion, continuous control, and discrete detection components of the model, we proposed the use of modern estimation (ME) techniques. For the situation assessment component of the model, we proposed the use of artificial neural networks (ANNs). For the decision-making, procedure selection, and procedure-effecting components of the model, we proposed the use of expert systems (ES) technology. We evaluated the applicability of each enabling technology to CSIM component implementation and design tool development.

We implemented a limited-scope version of the crew/system model to support a demonstration of its use in evaluating a display proposed for tactical SA. The task chosen was a rotorcraft tactical engagement in a limited air-to-air engagement. Major tasks facing the crew

were: identification friend or foe (IFF), prioritization of threat targets, and fire point selection (FPS). The analysis effort began with the development of both ownship and threat/friendly dynamic models, the development of a missile model, and the development of simplified radar models to provide relative range and velocity measurements to objects in the tactical area. The radar display was modeled on a Tactical Situation Display (TSD) for Rotorcraft Counter-Air Engagements (RCE) developed by HEL. The representation effort focused on modeling the information content of the display, the generation of derived states to feed a model of the crew's situation assessment function, the specification of a rule-base for decision-making, and the formalization of a procedure effector for guidance, control, fire-point selection, and other procedures. A nominal analysis was conducted with a baseline TSD display, and variants on the basic format were evaluated via model-based analysis. In addition, a number of decision aids were evaluated regarding their effectiveness in providing the crew with improved situation awareness (SA). In particular, four decision aids were evaluated: one to aid in the IFF function, one to aid in target prioritization, one for fire-point selection, and the last for TSD attention focusing. Of primary interest here was the degree of SA provided to the crew by changes in display format and/or changes in decision aid implementation.

6.2 Conclusions

The primary result of this study was a successful proof-of-concept demonstration of the model-based approach to evaluating situation awareness display requirements. The major study findings supporting this demonstration effort can be summarized as follows.

The specification of an overall crew/system model provided us with a development framework for evaluating the impact of display characteristics on crew awareness, and conversely, the impact of crew activity on system performance. It is an interactive framework that represents both the display characteristics and the crew procedures for dealing with the information in that display. It provides for an explicit representation of the crew's fundamental functions of information-processing (IP), situation assessment (SA), and decision-making (DM). It supports the development of objective metrics of performance and essay, and the subsequent evaluation of competing of display formats and decision aids.

The evaluation of enabling technologies focused on three areas: modern estimation (ME) models of the crew's IP function; artificial neural network (ANN) representations of the crew's essay functions; and expert system (ES) implementations of the crew's DM activities.

The modern estimation (ME) modeling work centered on the optimal control model (OCM) of the pilot/vehicle system. It is a standard tool for closed-loop evaluation of pilot performance under different monitoring assumptions and control options, and has been used

extensively in pilot/vehicle analysis. For this effort it was used to model the *front-end* processing functions of the pilot, specifically: attention allocation amongst the available displays, state estimation for the generation of continuous system states, and discrete event detection. In the modeling effort, it was assumed that the pilot had a close matched *internal model* of the external environment. However, in an extended configuration of the OCM, it is possible to evaluate task performance where there are distinct mismatches between the pilot's *internal model* of the external environment and the environment itself. This provides the potential for evaluation of display configurations and aids in cases where significant mismatches can be expected due to prior misconceptions on the part of the pilot regarding system dynamics.

The evaluation of artificial neural networks (ANNs) for representing the situation assessment behavior of the crew focused on the pattern recognition capabilities of ANNs. ANNs can provide general pattern recognition capabilities in the general state/event space generated by the CSIM information processor (IP), thus supporting the recognition of predefined *situational patterns* occurring over both space and time. Modeling of the pilot's SA function in this manner calls on a number of strengths afforded by ANNs. First, ANNs are particularly effective in implementing non-verbal pattern recognition functions which are algorithmically ill-defined and highly perceptual in nature. Second, ANNs can provide for considerable data compression with a many-on-few topological mapping from several state/event inputs to a few situational outputs. Third, the node normalization of ANN activation functions can be used to provide classification probabilities in a manner analogous to Bayesian discriminant analysis. Finally, an ANN model has the potential for modeling on-line unsupervised learning of tactically relevant patterns, and the sharpening of tactical skills that comes with repeated successful engagements. In this effort we focused on modeling the crew's recognition of threatening patterns, to support the IFF function.

The evaluation of expert system (ES) representations of the crew's decision-making behavior centered on data-driven procedural activities. We believe that the potential utility of an ES representation rests on four major factors. First, there is a natural mapping of the ES architecture to the crew DM functions; the ES data base represents the assessed situation, the ES rule base reflects the crew's procedure set, and the ES inference engine models the crew's DM algorithm(s). Second, an ES implementation can readily incorporate heuristics employed by the crew to *short-cut* extensive decision/procedure sequences, in accord with the stimulus-driven activity path postulated by several researchers. Third, ESs can implement dynamic updating of their knowledge base, to account for short-term trends peculiar to the mission phase or task. Finally, ESs can serve as a knowledge repository for implementation of the *upstream* crew functions of information processing (IP) and situation assessment (SA). In particular, ESs can

maintain a knowledge base of IP techniques and modes incorporated in the modern estimation (ME) formulation used to model the crew; likewise, ESs can characterize SA strategies and procedures represented in an ANN formulation of the SA function, via a specification of network topologies and weighting schemes used for different SA objectives during a mission. In this effort we focused on modeling the crewmember's decision functions for threat/friendly discrimination, target prioritization, and fire-point selection.

The proof-of-concept demonstration focused on a rotary-wing air-to-air engagement and provides: a basis for evaluating the requirements for problem setup; supports an objective evaluation of display formats and decision aids; and provides the foundations for a detailed evaluation of situational awareness during the engagement. The major findings of our demonstration effort can be summarized as follows:

- Display analysis and problem set-up is straightforward, and does not require an extensive procedural data base. Analysis of information content and format follows standard model-based approaches. Decision aid enhancements can be incorporated in a straightforward fashion through event generation and processing.
- Monitoring and control strategies on the part of the pilot can be explicitly defined. Attention allocation strategies used for processing of display information can be specified, as well as control strategies for guidance and fire-point selection.
- A variety of metrics can be evaluated with this approach. An overall metric representing the crew's SA can be computed both as a function of time and over the course of the engagement. In addition, metrics regarding state estimation accuracy can be computed, as well as engagement performance. Tracking of engagement progress can be effected via internal metrics. Enhancement-induced differences are reflected across metrics as well.
- The approach also provides a means of evaluating the specifics of the scenario (i.e., initial engagement geometry) on effectiveness of enhancements. In particular, enhancement effectiveness can be evaluated in low- versus high-tempo engagement scenarios.
- The effect of display format variations focused on the impact of display resolution and map scale range. The analysis showed that SA was effectively on the knee of the curve, in that a lower resolution TSD resulted in significant decreases in SA, whereas a higher resolution TSD yielded no effective improvement in SA.

- The effort demonstrated how use of an IFF display aid could result in a 50% increase in SA, as crewmember confidence on the display aid increased to 100%. A general linear progression in SA occurs with increasing confidence and use of the aid.
- The study with the priority display aid showed there is little effect under the normal scenario; however, when the number of threat/friendlies is doubled in a higher tempo scenario, a 15% improvement in SA is seen.
- The study of the FPS display aid also displayed little effect in the normal low-tempo scenario; however, a 40% improvement was seen in SA for the higher tempo scenario, with an approximately linear improvement in SA as confidence and use of the aid was increased.
- The study of the attention-focusing aid showed little effect when focusing was used to enhance a display. In fact, a slight negative effect was found due to the lack of information provided for low priority targets.

These results and others generated under this effort demonstrate that reasonably realistic activity traces can be generated with a fairly limited procedure rule base, and that the dynamic evolution of the pilot's situation assessment, decision-making, and task performance can be followed throughout an engagement. The results also show how simple metrics can be generated for internal model states related to SA and to external world states, to support evaluations of situation awareness and the effectiveness of display formats and decision aids.

Finally, the evaluation effort demonstrated how a number of engagement and crew variables affect engagement outcome. We can directly study the effects of: 1) initial engagement geometry; 2) sensor inaccuracies; 3) pilot-related limitations in sensory/perceptual processing; 4) internal state estimation strategy; 5) pattern recognition capabilities in situation assessment; and 6) decision-making procedure definitions. A wide variety of factors can be evaluated over a range of tactically relevant values.

6.3 Recommendations

We identified requirements for further development demonstration and validation of the display design/evaluation tool. Our recommendation is that the model-based method be developed into a prototype design/evaluation tool, to facilitate its introduction, use, and evolution in the user design community. A three-step prototype development and demonstration program under a follow-on effort is called for:

- **Simulation validation** of the integrated crew/system model, involving the design and conduct of a set of piloted real-time simulations of a selected set of tactical

engagements. A baseline scenario would be selected, a baseline cockpit configuration specified (displays and controls), and a limited number of display enhancement options chosen as the basic experimental factors. The objective would be to expand the scope of the model and scenario considered under the initial study, design and conduct a real-time simulation, and validate the model structure against empirical data. This would serve to generate a working data base for the crew's information processing, situation awareness, and decision-making functions; and it would serve to evaluate the model's ability to reliably account for the observed differences in crew SA and performance as a function of display enhancements. An illustration of the overall model validation protocol envisioned for this task is given in figure 6.1.

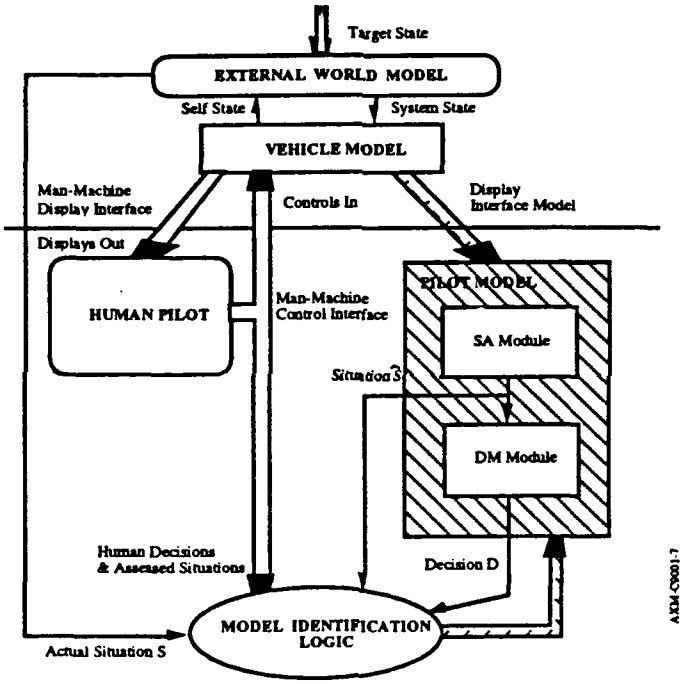


Figure 6.1: Model Validation Protocol

- **Design demonstration and validation** of the overall display design/evaluation technique with experiment design, engagement simulation, and crew SA analysis following along lines similar to that of the initial validation effort. The results of this step would allow us to validate the use of the display evaluation technique in a demonstration design exercise focusing on several candidate enhancement options, with their corresponding dependencies on alternative system concepts (e.g., new display formats, decision aids, etc.). This step would also serve to formalize the design/evaluation procedure, and identify basic steps in the protocol. The envisioned

design/evaluation procedure is illustrated in figure 6.2, in which an iterative design process is used for performance/SA optimization.

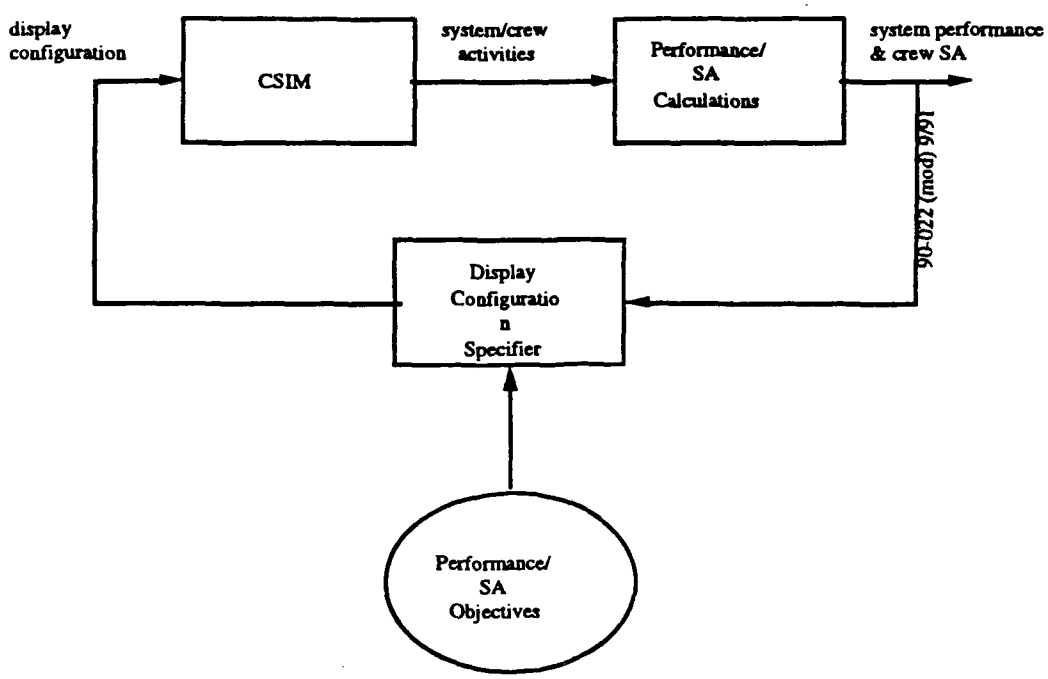


Figure 6.2: Design Process for Performance/SA Optimization

- **Prototype tool development and demonstration** of the computer-based design/evaluation tool: This would involve a significant software development effort to convert the research-oriented software into a computer-aided design package specialized for display design and evaluation. The effort would focus on the development of appropriate software structures and modules needed for transitioning this package to the design community. It would also involve a demonstration and evaluation of the package in a realistic design exercise.

7. REFERENCES

Anderson, J.A. and Rosenfeld, E. 1988. *Neurocomputing—Foundations of Research.* Cambridge, MA: The MIT Press.

Baron, S., Lancraft, R. and Zacharias, G. 1980. *Pilot/Vehicle Model Analysis of Visual and Motion Cue Requirements in Flight Simulation.* NASA Ames Research Center, CA. NASA Contractor Report 3312.

Baron, S. and Levison, W.H. 1973. *A Manual Control Theory Analysis of Vertical Situation Displays for STOL Aircraft.* NASA. CR-114620 (April).

Baron, S. and Levison, W.H. 1977. "Display Analysis with the Optimal Control Model of the Human Operator." *Human Factors*, Vol. 19, No. 5.

Baron, S., Zacharias, G., Muralidharan, R., et al. 1980. "PROCRU: A Model for Analyzing Flight Crew Procedures in Approach to Landing." *16th Annual Conf. on Manual Control.* Cambridge, MA.

Brown, J.S., Burton, R.R. and Bell, A.G. 1974. *SOPHIE: A Sophisticated Instructional Environment for Teaching Electronic Troubleshooting.* Bolt, Beranek and Newman, Inc., Cambridge, MA. 2790 (1974).

Brun, H.M. and Zacharias, G.L. 1986. *Model-Based Methodology for Terrain-Following Display Design.* Charles River Analytics Inc. R8603 (February).

Bryson, A.E.J., Ho, Y.C. 1975. *Applied Optical Control (Optimization, Estimation and Control).* Washington: Hemisphere Publishing Company.

Buchanan, B.G. and Feigenbaum, E.A. 1978. "DENDRAL and Meta-DENDRAL: Their Applications Dimension." *Artificial Intelligence*, Vol. II.

Carpenter, G.A. and Grossberg, S. 1988. "The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network." *IEEE Computer,* (March).

Cohen, M.A. and Grossberg, S. 1983. "Absolute Stability of Global Pattern Formation and Parallel Memory Storage by Competitive Neural Networks." *IEEE Transactions on SMC*, Vol. 13, No. 5.

Corker, K.e.a. 1986. *Development of an Advanced Task Analysis Methodology and Demonstration for Army-NASA Aircrew/Aircraft Integration.* Bolt Beranek and Newman, Inc., Cambridge, MA. 6124.

Corrigan, J. and Keller, K. 1989. "Pilot's Associate: An Inflight Mission Planning Application." *AIAA Guidance, Navigation and Control Conference.* Boston, MA.

Davis, R. 1983. "Reasoning from First Principles in Electronic Troubleshooting." *Int. J. Man-Machine Studies*, Vol. 19: 403-423.

Davis, R. 1984. "Diagnostic Reasoning from Structure and Behavior." *Artificial Intelligence*, Vol. 24: pp. 347-410.

Davis, R. 1987. *Knowledge-Based Systems: The View in 1986.* Cambridge, MA: The MIT Press.

Davis, R., Shrobe, H., Hamscher, N., et al. 1982. "Diagnosis Based on Descriptions of Structure and Function." *AAAI Proceedings.* Carnegie-Mellon University, Pittsburg, PA.

Dominessy, M.E., Lukas, J.H., Malkin, F.J., et al. 1991. "The Effect of Information Display Formats on Helicopter Pilots' Target Acquisition and Flying Performance." *Military Psychology*, : pp. 163-176.

Duda, R.O., Gaschnig, J., Hart, P.E., et al. 1978. *Development of the PROSPECTOR Consultation System for Mineral Exploration Inc.* SRI International Inc., Menlo Park, CA. 6415.

Fikes, R. and Kohler, T. 1985. "The Role of Frame Based Representation in Reasoning." *Communications of the ACM,* .

Forbus, K. 1987. "Interpreting Observations of Physical Systems." *IEEE Trans. on SMC*, Vol. 17, No. 3.

Forbus, K. 1988. *Qualitative Physics: Past, Present and Future—Exploring Artificial Intelligence*. Morgan Kaufman Publishers.

Forbus, K.D. *Qualitative Process Theory.* MIT AI Laboratory, Cambridge, MA. 1982.

Forbus, K.D. 1984. *Qualitative Process Theory*. Ph.D. Thesis. Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Department of

Fukeshima, K. 1988. "A Neural Network for Visual Pattern Recognition." *IEEE Computer,* (March): 65-75.

Gai, E.G. and Curry, R.E. 1976. "A Model of the Human Observer in Failure Detection Tasks." *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 6.

Gonsalves, P.G., Allen, S.M. and Caglayan, A.K. 1991. *An Alternative Concept for Aeroassisted Orbit Transfers*. Charles River Analytics Inc. R90222 (August).

Gonsalves, P.G. and Caglayan, A.K. 1992. *A Hybrid Neural Network/Expert System Approach to Multiple Target Recognition*. Charles River Analytics Inc., Cambridge, MA. Report No. R91121 (January).

Gonsalves, P.G. and Zacharias, G.L. 1989. *A Human Operator Model for Learning and Transfer of Training*. Charles River Analytics Inc. R8909 (September).

Greenstein, J.S. 1979. *Models of Human Decision Making in Multi-Task Situations*. University of Illinois. (October).

Grossberg, S. 1982. *Studies of Mind and Brain: Neural Principles of Learning, Perception, Development, Cognition, and Motor Control.* Boston, MA: Reidel Press.

Hecht-Nielsen, R. 1986. *Artificial Neural Systems Technology.* TRW Rancho Carmel AI Center, San Diego, CA. (June).

Hess, R.A. 1977. "Predictions of Pilot Opinion Ratings Using an Optimal Pilot Model." *Human Factors*, Vol. 19, No. 5: 459-475.

Hestenes, D. 1983. "How The Brain Works: The Next Great Scientific Revolution." *Maximum Entropy in Bayesian Spectral Analysis and Estimation Problems.* C.R. Smith, ed. Boston: Reidel Press.

Hopfield, J.J. 1982. "Neuronal Networks and Physical Systems with Emergent Collective Computational Abilities." *Proceedings of the National Academy of Sciences*, Vol. 79: 2554-2558.

Hopfield, J.J. 1984. "Neurons with Graded Response Have Collective Computational Properties Like Those of Two-State Neurons." *Proceedings of the National Academy of Sciences*, Vol. 81: 3058-3092.

Hopfield, J.J. and Tank, D.W. 1985. "Neural Computation of Decisions in Optimization Problems." *Biological Cybernetics*, Vol. 52: 141-152.

Hopfield, J.J. and Tank, D.W. 1986. "Computing with Neural Circuits: A Model." *Science*, Vol. 233: 625-633.

Hudlicka, E., et al. 1989. "Intelligent Situation Assessment and Response Aiding in Flight Emergencies." *Proc. AIAA Computers in Aerospace VII Conf.* Monterey, CA.

Human Engineering Laboratory. 1991. *Counter-Air Situation Awareness Display for Army Aviation*. U.S. Army Laboratory Command,. .

Kleinman, D.L. 1976. "Solving the Optimal Attention Allocation Problem in Manual Control." *IEEE Trans. on Automatic Control*, Vol. AC-21.

Kleinman, D.L. and Baron, S. 1971. *Analytic Evaluation of Display Requirements for Approach to Landing*. NASA. CR-1952 (November).

Kleinman, D.L., Baron, S. and Levison, W.H. 1970. "An Optimal Control Model of Human Response, Part 1: Theory and Validation." *Automatica*, Vol. 6.

Kleinman, D.L., Baron, S. and Levison, W.H. 1971. "A Control Theoretic Approach to Manned-Vehicle Systems Analysis." *IEEE Trans. on Automatic Control*, Vol. AC-16, No. December.

Kleinman, D.L. and Killingsworth, W.R. 1974. *A Predictive Pilot Model for STOL Aircraft Landing*. NASA. CR-2374 (March).

Kleinman, D.L. and Perkins, T. 1974. *Modelling the Human in a Time-Varying Anti-Aircraft Tracking Loop*. IEEE Trans. on Auto Control. AC-19.

Koshko, B. 1987. "Optical Bidirectional Associative Memories." *Soc. for Photo–Optical and Instrumentation Engineering: Image Understanding.*

Lancraft, R.E. and D.L., K. 1978. "A Comparison of Motor Submodels in the Optimal Control Model." *14th Annual Conference on Manual Control.* NASA.

Levison, W.H. 1971. *The Effects of Display Gain and Signal Bandwidth on Human Controller Remnant*. Wright-Patterson AFB, OH. AMRL-TR-70-93 (March).

Levison, W.H. and Baron, S. 1976. *Analytic and Experimental Evaluation of Display and Control Concepts for a Terminal Configured Vehicle*. Bolt, Beranek and Newman Inc. No. 3270 (July).

Levison, W.H., Baron, S. and Kleinman, D.L. 1969. "A Model for Human Controller Remnant." *IEEE Trans. on Man-Machine Systems*, Vol. 10.

Lizza, M.C.S. 1989. "Pilot's Associate: A Perspective on Demonstration 2." Wright-Patterson AFB, OH.

Milgram, P., van der Wijngaart, R., Veerbeek, H., et al. 1984. Multi-Crew Model Analytic Assessment of Landing Performance and Decision Making Demands." *Proc. 20th Annual Conference on Manual Control.* Moffett Field, CA. NASA Ames Research Center.

Mitchell, R.R. 1989. "Expert System and Air-Combat Simulation." .

Pattipati, K.R., Entin, E.E., Gully, S.W., et al. 1982. "A Normative-Descriptive Model of a Power System Dispatcher in Emergency Situations."

Pattipati, K.R., Ephrath, A.R. and Kleinman, D.L. 1979. *Analysis of Human Decision-Making in Multitask Environments*. University of Connecticut. UCONN EECS-TR-79-15.

Pattipati, K.R., Kleinman, D.L. and Ephrath, A.R. 1983. "A Dynamic Decision Model of Human Task Selection Performance." *IEEE Trans. on Systems, Man & Cybernetics*, Vol. 13(3): 145-156.

Pisano, A.D. and Jones, H.L. 1984. "An Expert Systems Approach to Adaptive Tactical Navigation." *First IEEE Conference on AI Applications.*

Rasmussen, J. 1983. "Skills, Rules and Knowledge: Signals, Signs and Symbolism, and Other Distinctions in Human Performance Models." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 12: 257-266.

Rosenblatt, F. 1962. *Principles of Neuro Dynamics*. Washington, D.C.: Spartan Books.

Ruck, D.W., Rogers, S.K., Kabrisky, M., et al. 1990. "The Multilayer Perceptron as an Approximation to a Bayes Optimal Discriminant Function." *IEEE Trans. on Neural Networks*, Vol. 1, No. 4: 296-.

Shortliffe, E.H. 1976. *Computer-Based Medical Consultations: MYCIN*. New York, NY: American Elsevier.

Stefik, M., Aikins, J., Balzer, R., et al. 1982. *The Organization of Expert Systems: A Prescriptive Tutorial*. Xerox. VLSI–82–1.

Taylor, E.C., Beebe, H.M., Goodman, H.S., et al. 1984. "Man and Machines-A Synergy of Tactical Intelligence." *Defense Electronics*, .

Visser, J. 1988. *PROCRU Simulation Results Compared with Metro II IN-Flight ILS Approach Data*. National Aerospace Laboratory NLR. NLR TR 87180L.

Wan, E.A. 1990. "Neural Network Classification: A Bayesian Interpretation." *IEEE Trans. on Neural Networks*, Vol. 1, No. 4: 303-305.

Wewerinke, P.H. 1981. "A Model of the Human Observer and Decision Maker." *Proc. 17th Ann. Conf. on Manual Control*. Los Angeles, CA. University of California.

Widrow, B. and Winter, R. 1988. "Neural Nets for Adaptive Filtering and Adaptive Pattern Recognition." *IEEE Computer*, (March): 25-39.

Zacharias, G.L. 1985. *Design and Performance Analysis for an Image-Flow Terrain-Following Guidance System*. Charles River Analytics Inc. R8501 (July).

Zacharias, G.L. 1985. *Modeling the Pilot's Use of Flight Simulator Visual Cues in a Terrain-Following Task*. Charles River Analytics Inc. R8505.

Zacharias, G.L. 1989c. *Situation Awareness Metric for Cockpit Configuration Evaluation*. Charles River Analytics Inc. R8910 (December).

Zacharias, G.L. and Baron, S. 1982. *A Proposed Crew-Centered Analysis Methodology for Fighter/Attack Missions*. Bolt Beranek and Newman Inc. 4866 (February).

Zacharias, G.L., Baron, S. and Muralidharan, R. 1982. "A Supervisory Control Model of the AAA Crew." *American Control Conference*.

APPENDIX A: MATHEMATICAL DESCRIPTION OF OPTIMAL CONTROL MODEL

A general block diagram of the OCM is given in figure A.1. The system portion (outside the dashed box) provides for representations of control interface dynamics, system (vehicle) dynamics, and any dynamics associated with the display interface. As shown, the two inputs to the system are the set of controls generated by the operator, and the system disturbances which act to perturb the overall system from equilibrium. The set of system outputs processed through the display interfaces is a multi-modality cue set driving the operator's various sensory systems. In the following two subsections, we describe the system and operator portions of the model in more detail.

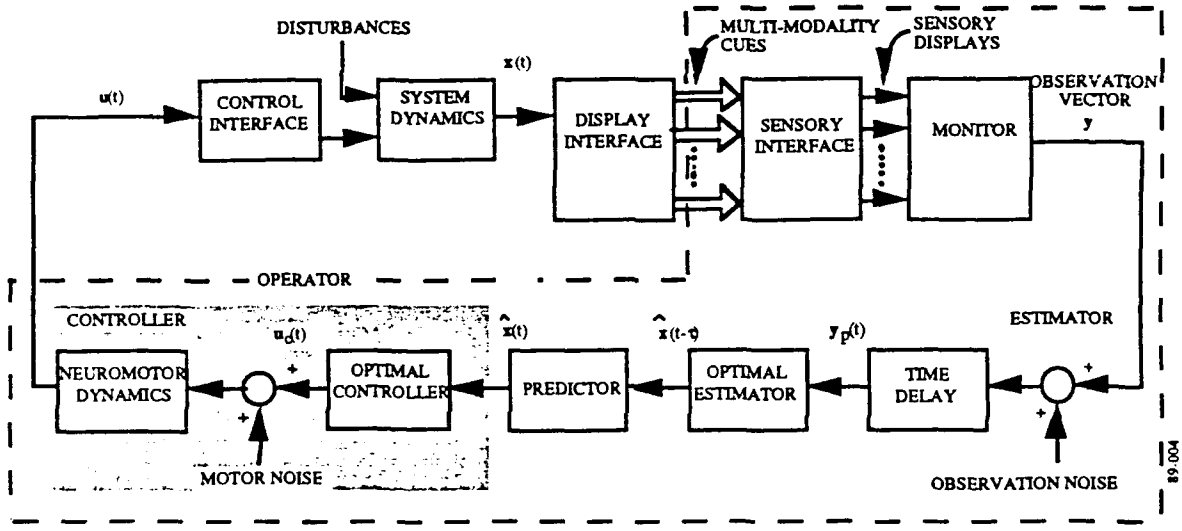


Figure A.1: Optimal Control Model of Human Operator/System

A.1 System Components of the OCM

System Dynamics:

As shown in the figure, the two inputs to the system are the set of controls (**u**) generated by the pilot and the system disturbances (**w**) which act to drive the overall loop. We assume that the vehicle is governed by a set of dynamics given by the following state equation:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{E}\mathbf{w}(t), \qquad \mathbf{x}(0) = 0 \tag{A.1}$$

where

$$\mathbf{x} \in R^n, \qquad \mathbf{u} \in R^m, \qquad \mathbf{w} \in R^p$$

Here $\mathbf{x}(\cdot)$ and $\mathbf{u}(\cdot)$ represent the system state and the control input, respectively. Also, $\mathbf{w}(.)$ is a vector-valued independent white Gaussian random process of covariance $\mathbf{W}$ disturbing the

vehicle. It is assumed that, the matrices **A**, **B**, and **E** have appropriate and compatible dimension (e.g., **A** is n by n, etc.).

The above system model includes all of the dynamics associated with all of the subsystems being controlled by the operator. In general, however, the system model will also include additional dynamics associated with other aspects of the task. For example, in a flight control task, we would include: a) any gust disturbances; b) any dynamics which limit the pilot's sensory capabilities in the given task; and c) any dynamics which might be used to approximate other system characteristics which cannot be expressed directly in terms of linear first-order vector-matrix equations. We discuss these points in the following paragraphs.

Insofar as terrain or gust models can be represented by rational noise spectra, they can be incorporated in the system model by first determining the appropriate shaping filter, which, when acting on white noise, generates the desired terrain or gust spectrum. By expressing this shaping filter in state-variable format, the system (A.1) may then be augmented to generate appropriate terrain/gust states which are driven by the white noise process vector **w**(t), through the disturbance input matrix **E**.

If the pilot's sensory dynamics are deemed relevant to understanding closed-loop performance in the given task, the dynamics may be expressed in state-variable form, and used to augment the system dynamics of (A.1). Thus, for example, if the pilot/vehicle response bandwidth is expected to be influenced by the pilot's vestibular dynamics, then these dynamics can be accounted for directly, by appropriate augmentation of the system state equations.

System dynamics which, after linearization, are not directly expressible in the form of (A.1) may be included in the system description by first finding a suitable state-variable approximation and then augmenting (A.1) with this approximation. Pure time delays, in particular, are conveniently handled by this approach. Once an appropriate Pade' filter approximation to the delay is found, the associated state-variable dynamics can be directly included in the system dynamics of (A.1).

In summary, the system (A.1) not only includes the explicit dynamics of the various subsystems involved, but also the implicit dynamics associated with the disturbance spectra, the relevant sensory dynamics of the operator, and any additional approximations deemed necessary for accurate system modeling.

Display Interface:

The display interface provides a means for transforming the system state variables and the operator's control actions into a display *vector* which represents that set of all information

available to the operator. The components of the display vector are assumed to be linear combinations of the state and control variables, and are defined by the following r-dimensional vector equation:

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{A.2}$$

where **C** and **D** may be time-varying (piece-wise constant) to account for changes in the quantities being displayed or *observed.* Generally, the display vector includes *displays* from many sensory modalities, such as visual, vestibular, proprioceptive, tactile, or auditory. Our concentration in this study will be on the visual modality.

In general, the processing provided by the operator's sensory systems requires a model which involves not only a linear transformation of the system state (as in (A.2)), but also a dynamic transformation which accounts for any important sensory processing dynamics (e.g., vestibular dynamics). As just noted, this latter modeling requirement is implemented by assigning the sensory dynamics to the set of overall system dynamics, and appropriately augmenting the state equation of (A.2).

A.2 Operator Components of the OCM

The basic assumption underlying the Optimal Control Model of the operator is that the well-trained, well-motivated human controller will act in a near optimal manner, subject to certain internal constraints that limit the range of his behavior, and also subject to the extent to which he understands the task objectives. When this assumption is incorporated in the OCM framework and when appropriate limitations on the human are imposed, the structure shown in he bottom half of figure A.1 results. In discussing this structure, it is convenient and meaningful to view this model as being comprised of the following:

1. An equivalent *perceptual* model that translates the displayed variables $\mathbf{y}(t)$ into noisy, delayed, *perceived* variables denoted by $\mathbf{y}_p(t)$
2. An equivalent *motor* or output model that accounts for *bandwidth* limitations (frequently associated with neuromotor dynamics) of the human, and the inability to generate noise-free controls
3. An *information processor,* consisting of an optimal (Kalman) estimator and predictor that generates the minimum-variance estimate $\hat{\mathbf{x}}(t)$ of $\mathbf{x}(t)$
4. A set of *optimal gains* chosen to minimize a quadratic cost functional that expresses the task requirements

We now discuss these model components in greater detail.

Perceptual Model:

Limitations on the operator's ability to process information displayed to him are accounted for in the *equivalent* perceptual model. This model translates the displayed variables y into delayed, *noisy* perceived variables $\mathbf{y}_p$ via the relation

$$\mathbf{y}_p(t) = \mathbf{y}(t-\tau) + \mathbf{v}_y(t-\tau) \tag{A.3}$$

where τ is an *equivalent* perceptual delay and $\mathbf{v}_y$ is an *equivalent* observation noise vector having covariance $\mathbf{V}_y$.

The various internal time delays associated with visual, central processing, and neuromotor pathways are combined and conveniently represented by this lumped equivalent perceptual time delay τ. Typical values for this delay are 0.2 +/- .05 sec (Kleinman and Baron (1971)).

The observation noise $\mathbf{v}_y$ is included to account for the operator's inherent randomness, due to random perturbations in human response characteristics, errors in observing displayed variables, and attention-sharing effects which limit the operator's ability to accurately process all the cues simultaneously available to him. In combination with the motor noise model (described below), the observation noise model provides a convenient and accurate means of modeling operator remnant and accounting for random control actions. Each component of the noise vector $\mathbf{v}_y$ is assumed to be a random process which is linearly independent of other such noise processes and of external disturbances inputs to the system.

For manual control situations in which the displayed signal is large enough so that visual resolution (*threshold*) limitations are negligible, the auto covariance of each observation noise component appears to vary proportionally with mean-squared signal level. In this situation, the auto covariance of the noise associated with the ith display component may be represented as

$$V_{yi} = \pi P_{yi} \sigma_{yi}^2 \tag{A.4}$$

where σ_{yi}^2 is the variance of the ith display variable, and P_{yi} is the *noise/signal ratio* for the ith display variable, which has units of normalized power per rad/sec. Numerical values for P_{yi} of 0.01 (i.e., -20 dB) have been found to be typical of single-variable control situations (Levison, et al. (1969), Kleinman, et al. (1970)).

The perceptual model defined by (A.3) and (A.4) applies to *ideal* display conditions, in which the signal levels are large with respect to both system-imposed and operator-associated thresholds. To account for threshold effects we let the auto covariance for each observation noise process be

$$V_{yi} = \pi P_{yi} \left\{ \frac{\sigma_{yi}^2}{K^2(\sigma_{yi}, a_i)} + \sigma_{oi}^2 \right\} \tag{A.5}$$

where the subscript i refers to the ith display variable. The quantity $K(\sigma_{yi}, a_i)$ is the describing function gain associated with a threshold device given by

$$K(\sigma,a) = \frac{2}{\sqrt{\pi}} \int_{-}^{\frac{a}{\sigma\sqrt{2}}} e^{-x^2} dx \tag{A.6}$$

where "a" is the threshold and σ is the standard deviation of the *input* to the threshold device. The net result of this type of describing function model is to increase the observation noise covariance as the display signal variance becomes smaller relative to the threshold. The quantity σ_{oi}^2 is a *residual* noise term which is introduced to account for performance degradation that arises when an explicit zero-error reference is lacking (Levison (1971)).

The sources of these threshold effects depend on the particular task being modeled. They may be associated with the system display implementation, for example due to resolution limitations on a display screen. Or, they may be associated with the operator's sensory limitations, such as one might identify with visual acuity thresholds.

One additional factor which tends to increase the observation noise (present on any given display variable) is the operator's attention-sharing limitations. Because the numerical value associated with the operator's noise/signal ration (P_y) has been found, in single display situations, to be relatively invariant with respect to system dynamics and display characteristics, we associate this parameter with limitations in the operator's overall information-processing capability. This forms the basis for a model for operator attention-sharing where the amount of attention paid to a particular display is reflected in the noise/signal ratio associated with information obtained from that display (Baron and Levison (1973)). Specifically, the effects of attention-sharing are represented as

$$P_{yi} = P_y / f_i \tag{A.7}$$

where P_{yi} is the noise/signal ratio associated with the ith display. When attention is shared among two or more displays, f_i is the fraction of attention allocated to the ith display, and P_y is the noise/signal ratio associated with full attention to the task.

Motor Model:

Limitations on the operator's ability to execute appropriate control actions are accounted for in the motor model, which is composed of a white motor noise source and a first-order filter with time constant τ_n. This model translates the *commanded* controls, $\mathbf{u_c}$, into the output control actions u via the following relation:

$$\begin{aligned}\dot{\mathbf{u}} &= (1/\tau_n)\,\mathbf{u}(t) + \mathbf{u_a} + \mathbf{v_u}(t) \\ &= \mathbf{u_c}(t) + \mathbf{v_u}(t)\end{aligned} \tag{A.8}$$

where $\mathbf{v_u}$ is an *equivalent* motor-noise vector with covariance $\mathbf{V_u}$, and where $\mathbf{u_c}(t)$ is generated to provide quadratic cost minimization, as described below. In laboratory tracking tasks with optimized control sticks, the motor time constant has been set to a value of about 0.1 sec. For more realistic flight control situations, however, this bandwidth limitation may be overshadowed by the system dynamics and flight control objectives, so that larger values may be more appropriate.

The neuro-motor noise vector of (A.8) is provided to account for random errors in executing intended control movements, and, in addition, to account for the fact that the operator may not have perfect knowledge of this own control activity. The motor noise is assumed to be a white noise, with auto covariance that scales with the control rate variance, i.e.,

$$V_{ui} = \pi\, P_{ui}\, \sigma_{ui}^2 \tag{A.9}$$

Previous studies have found, typically, that a value for P_u of 0.0001 (i.e., a "motor noise ratio" of -40 dB) yields good agreement with experimental results (Lancraft and D.L. (1978)).

Augmented System Equations:

Combining equations (A.1), (A.2), (A.3), and (A.8) we obtain the augmented system,

$$\begin{aligned}\dot{\mathbf{z}}(t) &= \mathbf{A_0 z}(t) + \mathbf{B_0 u_c}(t) + \mathbf{E_0 w}_1(t), \qquad \mathbf{z}(0) = 0 \\ \mathbf{y}\ (t) &= \mathbf{C_0 z}(t) \\ \mathbf{y}\ _p(t) &= \mathbf{y}(t-\tau) + \mathbf{v_y}(t)\end{aligned} \tag{A.10}$$

where

$$\mathbf{A}_0 = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ 0 & 0 \end{bmatrix}, \quad \mathbf{B}_0 = \begin{bmatrix} 0 \\ \mathbf{I} \end{bmatrix}, \quad \mathbf{C}_0 = [\, \mathbf{C} \quad \mathbf{D} \,],$$

$$\mathbf{E}_0 = \begin{bmatrix} \mathbf{E} & 0 \\ 0 & \mathbf{I} \end{bmatrix}, \quad \mathbf{z}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{bmatrix}, \quad \mathbf{w}_1(t) = \begin{bmatrix} \mathbf{w}(t) \\ \mathbf{v}_d(t) \end{bmatrix} \tag{A.11}$$

The covariance of the noise process $\mathbf{w}_1(t)$ is denoted $\mathbf{W}_1$.

Regulator and Estimator Equations for the OCM:

The operator is assumed to use a steady state control strategy based on the task objective. The strategy is assumed to be optimal for a quadratic cost criteria J derived from the task objectives and given by,

$$J(\mathbf{u}) = E\left\{\mathbf{x}'\mathbf{Q}_x\mathbf{x} + \mathbf{y}'\mathbf{Q}_y\mathbf{y} + \mathbf{u}'\mathbf{Q}_u\mathbf{u} + \mathbf{u}_c'\mathbf{G}\mathbf{u}_c'\right\} \tag{A.12}$$

It can be shown that the OCM is comprised of two separate modules ([[Kleinman, 1971 #348]): an optimal regulator for the cost criteria J, fed by an optimal estimator for $\mathbf{z}(t)$. In turn, the optimal estimator for $\mathbf{z}(t)$ cascades an optimal estimator for $\mathbf{z}(t-\tau)$ with an optimal predictor to compensate for the delay τ. We describe below in detail the design of each of these blocks.

The selection of the weightings $\mathbf{Q_x}$ = diag [$\mathbf{q_{xi}}$], $\mathbf{Q_y}$ = diag [$\mathbf{q_{yi}}$], $\mathbf{Q_u}$ = diag [$\mathbf{q_{ui}}$] and G = diag [$\mathbf{g_i}$] in J ($\mathbf{u_i}$) is a nontrivial step in applying the OCM. The most commonly used method for selecting reasonable *a priori* estimates for the output weightings ($\mathbf{Q_y}$) is to associate them with allowable deviations in the system variables; this method has been described in several applications of the OCM (see, for example, Kleinman (1976)). The state and control related weightings ($\mathbf{Q_x}$, $\mathbf{Q_u}$, G) may be chosen in a similar fashion or they may be picked to yield a desired response bandwidth.

Optimal Regulator:

The cost functional in (3.12) can be rewritten as

$$J(\mathbf{u}_c) = E\left\{\mathbf{z}_0'\mathbf{Q}_0\mathbf{z}_0 + \mathbf{u}_c'\mathbf{G}\mathbf{u}_c'\right\} \tag{A.13}$$

where

$$\mathbf{Q}_0 = \begin{bmatrix} \mathbf{Q}_x + \mathbf{C}'\mathbf{Q}_y\mathbf{C} & \mathbf{C}'\mathbf{Q}_y\mathbf{D} \\ \mathbf{D}'\mathbf{Q}_y\mathbf{C} & \mathbf{Q}_u + \mathbf{D}'\mathbf{Q}_y\mathbf{D} \end{bmatrix} \tag{A.14}$$

The optimal regulator gain L is given by

$$\mathbf{L} = \mathbf{G}^{-1}\mathbf{B}_0'\mathbf{K}_0 \tag{A.15}$$

where K_0 is the unique, symmetric, positive semi-definite solution to the algebraic Riccati equation,

$$\mathbf{A}_0'\mathbf{K}_0 + \mathbf{K}_0\mathbf{A}_0 + \mathbf{Q}_0 - \mathbf{K}_0\mathbf{B}_0\mathbf{G}^{-1}\mathbf{B}_0'\mathbf{K}_0 = 0 \tag{A.16}$$

Partitioning $\mathbf{L}$ as $\mathbf{L} = [\mathbf{L}_* | \tau_n^{-1}]$, we see that the motor response time constant is given by τ_n. The cost penalty $\mathbf{G}$ is chosen to be such that the gain on the $\mathbf{u}$ component of $\mathbf{z}$ is τ_n^{-1}.

Optimal Estimator:

This block computes the optimal estimate $\mathbf{p}(t)$ of the delayed state $\mathbf{z}(t-\tau)$. Based on the dynamics of (A.10), the optimal estimator gain $\mathbf{F}$ is given by

$$\mathbf{F} = \Sigma\, \mathbf{C}_0'\, \mathbf{V}_y^{-1} \tag{A.17}$$

where Σ is the symmetric, positive semi-definite solution to the following Riccati equation:

$$\mathbf{A}_0\Sigma + \Sigma\mathbf{A}_0' + \mathbf{E}_0\mathbf{W}_1\mathbf{E}_0' - \Sigma\mathbf{C}_0'\mathbf{V}_y^{-1}\mathbf{C}_0\Sigma = 0 \tag{A.18}$$

where

$$\mathbf{W}_1 = \begin{bmatrix} \mathbf{E}\,\mathbf{W}\,\mathbf{F}' & 0 \\ 0 & \mathbf{V}_u \end{bmatrix} \tag{A.19}$$

The estimator structure is then given by

$$\dot{\mathbf{p}}(t) = \mathbf{A}_0\mathbf{p}(t) + \mathbf{B}_0\mathbf{u}_c(t) + \mathbf{F}\nu(t), \quad \mathbf{p}(0) = 0 \tag{A.20a}$$

where the innovations process $\nu(t)$ is given by

$$\nu(t) = \mathbf{y}_p(t) - \mathbf{C}_0\mathbf{p}(t) \tag{A.20b}$$

Optimal Predictor:

The optimal estimate $\mathbf{p}(t)$ of $z(t-t)$ at time t is passed through a predictor block to obtain an optimal estimate $\hat{\mathbf{z}}(t)$ of $\mathbf{z}(t)$. The predictor block is defined by

$$\hat{\mathbf{z}}(t) = e^{\mathbf{A}_0\tau}\mathbf{p}(t) + \int_{t-\tau}^{t} e^{\mathbf{A}_0(t-s)}\mathbf{B}_0\mathbf{u}_c(s)\, ds \tag{A.21a}$$

The feedback control is then given by

$$\mathbf{u}_c(t) = \mathbf{L}\,\hat{\mathbf{z}}(t) \qquad \text{(A.21b)}$$

As noted above, the tandem of predictor and estimator generate a minimum variance estimate of the system state. As such, they (linearly) compensate for any time delays or noises introduced by the system and/or the operator. These elements incorporate *perfect* models of the dynamic environment, including models of the vehicle and gust spectrum, and models of the operator's own sensory dynamics. Thus, model predictions are appropriate for operators that are knowledgeable about the system characteristics, and about their own sensory capabilities and limitations.

This, then, provides a description for the elements of the Optimal Control Model of the human operator. It should be emphasized that the parameter values that must be specified by the user correspond to the human and system limitations that constrain overall operator/system performance. With these limitations as the constraints within which performance is produced, the model predicts the best that the operator can do, on the given task.

The OCM has been validated against experimental data for a variety of tasks, and detailed results may be found in the literature. In one of the earliest validation studies involving closed-loop tracking, it was found that error scores, describing function, and operator randomness were affected by multiple-task requirements all in the manner predicted by the model (Levison, et al. (1969)). Gai and Curry (1976) used the OCM information-processing structure to analyze failure detection in a simple laboratory task, and in an experiment simulating pilot monitoring of an automatic landing approach.